I/O NEWS

Software Survey
User Notes
New Products

THE OFFICIAL PUBLICATION OF THE INTERNATIONAL ASSOCIATION OF CROMEMCO USERS

Volume Five, Number Three

Single Copy Price $10.00

CS 200:
The Black Box Is Back

Cromemco has announced a new rack-mount computer system designed specifically for industrial, laboratory, and military markets. The new System 200 (model CS-200) is a derivative of Cromemco's classic System 2 (model CS-2), but with many new features and benefits.

The System 2 was originally designed to be used in an 8-bit CPU environment with up to 64K of memory. Now that Cromemco systems have moved to a 32-bit CPU and can support up to 16 megabytes of memory, the time has come to bring the System 2 in step with the rest of Cromemco's product line.

The biggest difference between the System 2 and the System 200 is in the power supply (see Table 1). The System 200 has a larger, more efficient supply with 50% more capacity than that of the System 2. This gives the System 200 supply a full 450 volt-ampere output capacity. A new transformer and rectifier design has been incorporated into the supply to reduce supply dissipation and improve the supply's voltage regulation over its full operating range. With its new 450 volt-amp capacity, the System 200 can accommodate virtually any combination of Cromemco circuit cards in its 21-slot card cage.

Since today's high-performance circuit cards dissipate more power than earlier cards, not only is a larger more rugged power supply required, but additional cooling is required as well. While the CS-2 had a single cooling fan, the CS-200 has three high pressure blowers. In addition, the baffling of the system has been improved to achieve more uniform air flow over the heat-dissipating components of the system.

As systems have increased in capability, there has been an increased need for more communication channels. These channels might be used to support multiple users in a multi-user system or to support telecommunication channels or peripheral I/O. The back panel of the CS-200 can now accommodate 20 I/O connectors, double the number of the CS-2.

Continued on page 8

APPGEN: A Product Review

by Robert Brown

APPGEN, Introduction

APPGEN is a fourth generation language applications generator. It was specifically created to provide software developers a method to CREATE and MAINTAIN business (accounting) applications. It is now available for Cromemco Systems under UNIX System V.

APPGEN, Claims

I spent a week in Houston at Software Express, learning about APPGEN along with a dozen other individuals. APPGEN really interested me, if for no reason than some of the claims made for their software. Software Express says that with APPGEN, developers can:

Continued on page 9

Cromemco Q & A

Since its inception in 1974, Cromemco has been a leader in the micro and supermicro computer markets. Products developed range from single boards running under the CP/M operating system up through integrated systems that utilize dedicated 32-bit processors and run under the UNIX operating system. To remain competitive, Cromemco is constantly integrating new technologies in the most cost effective manner available. Over the last 12 years this has allowed Cromemco to consistently produce high performance products at the lowest possible cost.

The combination of existing and new technology into a single product often constitutes a difficult design feat. It is small wonder then that even the most technically knowledgeable observer of

Continued on page 11

ANNOUNCING!!

dBIII Compiler

for 68000 Cromix-Plus

dBIIICompiler from WordTech Systems is a powerful dBase III compiler. Compiled dBase II programs run without the presence of dBase III. The compiler gives the programmer the ability to generate machine-efficient, stand alone, effectively 'encrypted' programs. Generally, programs will execute faster when compiled and require less space.

dBIIICompiler is fully terminal independent using the /etc/termcap file. Also, the system supports both dBase II as well as dBase III files with full multi-user capabilities including record level locking. In addition, all printed output is inherently spooled. Look for our new **UNIX SYSTEM V** version to be released second quarter '86.

Contact your Cromemco dealer for more information. If you don't have a Cromemco Dealer, contact Software Standards for the name of the dealer nearest you.

Software Standards, Inc.

6191 Choctaw • Baton Rouge, La. 70805
504/355-8024

Industrial grade S-100 boards don't cost more, anymore!

Dual Systems has just reduced the prices on its industrial grade IEEE-696/S-100 boards for 16-bit microprocessor systems. Whether running under UNIX or any other operating system, these boards bring high performance and years of field-proven reliability to your computing environment. Each board is rigorously tested and burned-in for 168 grueling hours. If it can't bear the heat, it won't bear our name.

High Performance System Boards

Model WDC-SMDX The WDC-SMDX Hard Disk Controller is specially designed for high throughput in large, heavily-loaded multi-user UNIX systems. It offers 16-bit throttled DMA data transfers and disk transfers up to 15 Mb/sec. Also features dual-ported, full-track, look ahead cache, and on-board microprocessor. Interfaces with one or two SMD drives. $2195.

A 10 Mb/sec. version is also available. $1995.

Model S104-DMA The most advanced, intelligent, 4-port serial I/O board available for the IEEE-696/S-100 bus, this module features 256 bytes of FIFO buffer for input characters, and provides DMA transfers for output. A built-in 8085A processor greatly reduces system overhead. $595.

Model DMEM Features 256K bytes of memory and either 8- or 16-bit data paths. 24-bit addressing, and parity checking on each byte. DMEM has no S-100 wait states. $595.

Model EPROM Capable of either 8- or 16-bit data transfers, this 32/64K EPROM offers the versatility of running with 68000, Z-8000, 8086, 16000, and other 16-bit processors. It accepts industry-standard 2732 and 2716 EPROMs. 64K RAMS may be mixed with 2716 EPROMs for use as a RAM/EPROM board. $295.

Model CPU-68000M High-performance CPU board with 16-bit data path, 10 MHz CPU operation, and MC68451 MMU for multi-tasking applications. $1195.

Model CMEM This non-volatile CMOS memory board provides easy-to-use 8- or 16-bit data paths and 32K bytes of memory with dynamically movable write/protect window. On-board lithium battery holds data for 3-10 years with power off. $725.

Model EMEM-1MB Features 1 megabyte of memory and either 8- or 16-bit data transfers. 24-bit addressing and parity error checking on each byte. Runs at 6 MHz for both 8- and 16-bit systems. No S-100 wait states except during refresh. $1995.

Model TCON Nine-track tape controller supports industry-standard IBM-ANSI formatter interface and DMA on read and write to tape. 24-bit addressing on DMA transfers; 8-bit data on DMA transfers. 512-byte FIFO data buffer. Supports tape densities up to 6250 BPI. $1250 with software drivers; $850 alone.

Data Acquisition and Control Boards

Model CLK-24C Clock calendar features a LSI CMOS chip and on-board, long-life lithium battery. $295.

Model AIM-12 A highly reliable A-to-D converter with 35 microsecond maximum conversion time, 12-bit resolution and accuracy, and 32 channels single-ended/16 channels differential. $725.

Model AOM-12 This D-to-A converter offers I/O-mapped port address, 12-bit ± ½ L.S.B. accuracy (0-70 °C), and voltage outputs of 0 to 10 volts, ±5 volts, and ±10 volts. $675.

Model VIC 4-20 Converts voltage outputs from AOM-12 into four separate 4-20MA current outputs. Module also provides overvoltage protection on all current output, plus transient protection per ISA standards. $600.

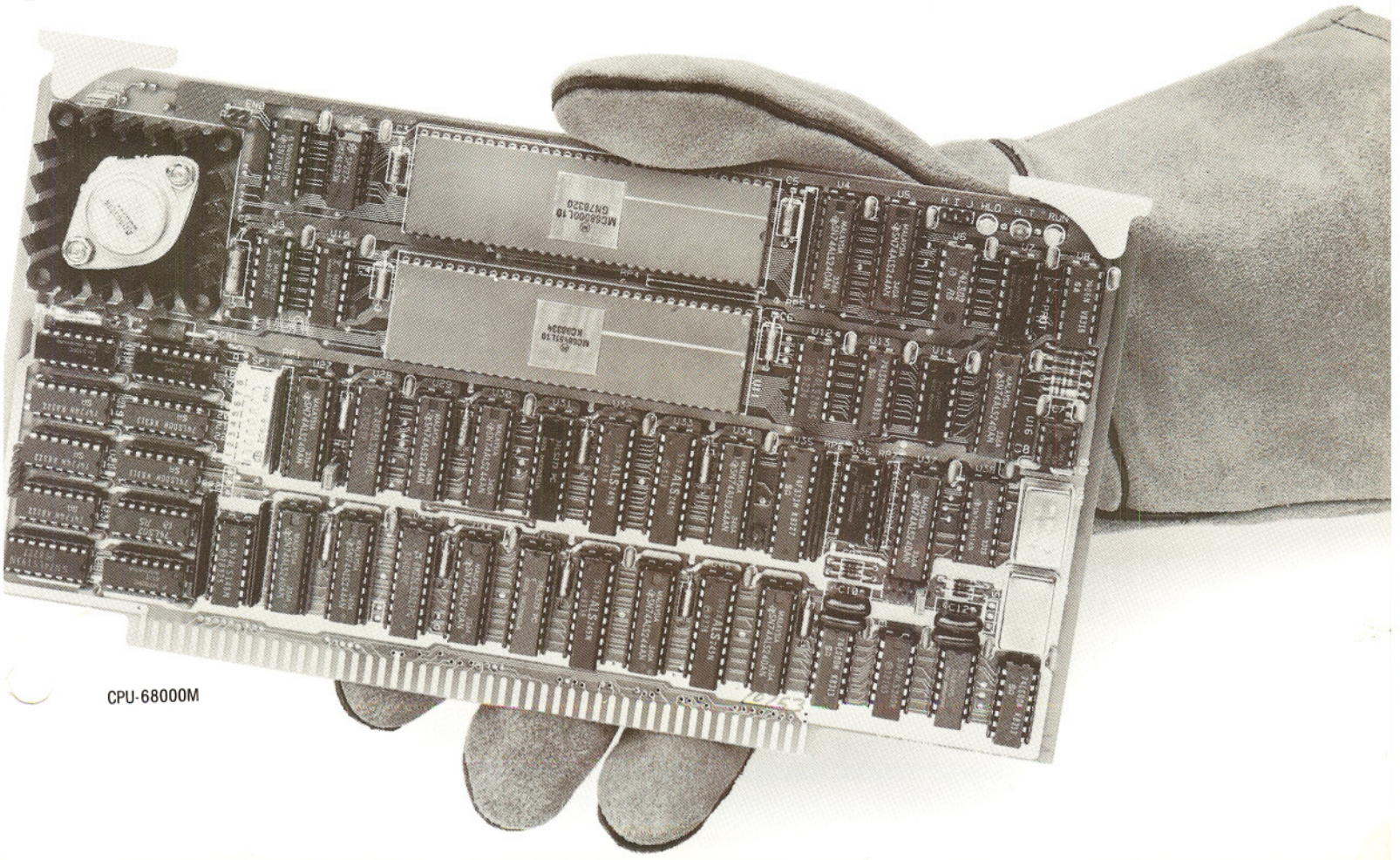
Dealer and volume discounts available.

For more information, call (415) 549-3854.

Dual Systems Corp., 2530 San Pablo Avenue, Berkeley, CA 94702

THERE'S ONLY ONE

DUAL



CPU-68000M

table of contents

March/April, 1986 Volume Five, Number Three

I/O NEWS *The Official Publication of The International Association of Cromemco Users is available through membership in the association. Editorial and advertising policies are designed for the enlightenment of the members in regard to new uses for, and developments of, Cromemco products and other products compatible with Cromemco systems.*

COVER FEATURES

ARTICLES & FEATURES

DEPARTMENTS

I/O NEWS [ISSN 0274-9998] is published bi-monthly by The International Association of Cromemco Users (a California corporation), P.O. Box 17658, Irvine, CA 92713. General Offices are at 34021 Granada #B, Dana Point, CA 92629. Telephone: (714) 661-9764. Second Class Postage Paid at Dana Point and Santa Ana, CA. **POSTMASTER: Send address changes to I/O NEWS, P.O. Box 17658, Irvine, CA 92713.**

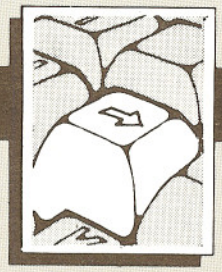
Subscriptions to I/O NEWS are entered with membership in The IACU. Yearly memberships may be purchased for $42 (U.S. delivery address), $49 (delivery address in Canada or Mexico), and $60 (other international delivery addresses). Contact IACU for multi-year membership rates. Back issues of I/O NEWS are available for $10.00 per issue. Please note: all prices are in U.S. dollars.

Return postage must be included with all manuscripts and photos submitted if they are to be returned. The IACU and I/O NEWS accept no responsibility for the return of unsolicited materials.

William E. Jaenicke
Editor and Publisher

Lisa B. Jaenicke
Business Manager

Art Direction: Howard Millman
Printing: Hallmark Litho
Typography: Western Outdoors



INPUT...

Dear Bill & Lisa:

You are both doing great jobs with I/O NEWS! I like the "theme" approach of Vol. V, No. 2 (i.e. TELECOMMUNICATIONS). Let's see an issue on "SPREADSHEETS," "STATISTICS" (i.e., statistical analysis packages usable with CROMEMCO). I have worked with three CP/M statistical packages and run them directly off CDOS. So far, I only have demo packages. One of these, SYSTAT (written by Dr. Leland Wilkinson) was originally developed on a Cromemco system. SYSTAT 3.0 is now available for the Apple McIntosh. However, he claims sales would not justify porting it to 68000 Cromix ... what a shame! Maybe a little subsidy from Cromemco could get this powerful package ported and available for direct Cromix-Plus use.

In regards to the IACU Software Resource Guide: how about making a membership option to receive updates? How about Cromemco making such a factory option for subscribers?

Thanks,
George A. Collier Jr.
Durant, Oklahoma

See OUTPUT for comments.
Ed.

Dear Editor:

That was a great program that Rick Dhaenens provided (CFSU, in Volume V, Number One of I/O NEWS). After I typed in the program, it naturally didn't work. I fixed most of the obvious errors, and still, problems.

I found some bugs to report.

In the function readiblk(), buffer is declared as int buffer []; it should be char buffer[].

In the function ls(cc,cv), where the case is is__ordin the number should be %lld and not %llld. By the way, is there a way you can make lower case L's stand out from ones?

My major problems came from cfsu.h not being listed. Apparently structs.h for 68000 C has some errors. Many integers are declared as unsigned when they should be short. Also, Rick must have several of his own definitions, as my structs.h has different member names and no definitions at all for DIRENTRY, SUPERBLK, INODE, NUMDIRS, BLOCKSIZ and BLOCK.

I finally have the program working and it's great. Just wanted to drop a line to let you know about the problems. I'm ready to write UFSU next!

Jack Destes
W.E.S.T.
Carmichael, California

Thanks for the info, and sorry about cfsu.h being absent from the program listing — that was an oversight. And we hope to see your version of UFSU when it's ready. As for distinguishing lower case "l" from the digit "1", well, that's something we have to live with until the designers of typographic fonts realize how important that distinction is in a computer program. By the way, all programs which appear in I/O NEWS are available (source and compiled). Contact I/O NEWS to arrange for delivery via modem or disk.
Ed.

Dear Bill,

After reading Volume V, Number 2 of I/O NEWS I felt I had something to offer your readers. Enclosed is a 'C' routine for general purpose input demonstrating termcap usage and the Julian Date routine contributed by Robert Brown. Also I have included termcap information that I have obtained from other sources. I have much more termcap information, but I don't know how privileged it is. If you could get me a name, or number at Berkeley, I would like to find out how much of this information I could turn over to you for publication.

I have successfully compiled and used termcap under CDOS, Cromix 11.27 and Cromix-Plus. I can also recommend successful results with **Faircom c-tree ISAM** (Telephone: (314) 445-6833). I only recommend this ISAM because source code is available. 'C' is only portable if all libraries are available in source code.

Hardware Tec Tip: Cromix-Plus drivers no longer check for ready signal. The driver checks only to see if an acknowledge is outstanding. Jumpering strobe to acknowledge will not allow the printer to appear busy to the computer.

Cromix-Plus Driver Tip: QSLPT drivers contain a character translation table which makes the CLQ (Smith Corona TP1) work well. This translation quickly has a better idea on NEC printers working with microjustification. I suggest using the QTTY driver and setting appropriate modes in the /etc/startup.cmd. Don't forget to set baud and -PA. Also make sure entry is 0 in the /etc/ttys table, otherwise the slpt might try to logon, unexpected and very humorous results may be observed.

Sincerely,
Colin "Soup" Campbell
Fairbanks, Alaska

File TEST.H Sunday, April 6, 1986 13:04:51

```
#include <stdio.h>
#define isdigit(c) ((c)<='9' && (c) >='0')
char    *BC,*CM,*UP,*CL,*CE,*CD;

char    buf[512];
char    *getenv(), *tgetstr();
char    *rdchar();
char    *tgoto();
char    cmbuff[100];
char    *x;
char    *tgout;
char    outfile[50];
```

File INIT.C Sunday, April 6, 1986 13:08:19

```
init_pgm(argc,argv)
int argc;
char **argv;
{
char    *p;
int     rc;

        if (argc < 2)
                p = getenv("TERM");
        else
                p = argv[1];
        rc = tgetent(buf,p);
        x=cmbuff;
        CD=tgetstr("cd",&x);    /* clear to end of display *
        CE=tgetstr("ce",&x);    /* clear to end of line */
        CL=tgetstr("cl",&x);    /* clear entire screen */
        BC=tgetstr("bc",&x);    /* backspace */
        UP=tgetstr("up",&x);    /* up n lines */
        CM=tgetstr("cm",&x);    /* cursor motion */
}
```

File INPUT.C Monday, April 14, 1986 18:34:39

```
#define TESTINPUT
#ifdef TESTINPUT
#include "test.h"
char    ans[50];
char    temp[20];
int     ans_length;
int     m_day;

main(argc, argv)
int argc;
char **argv;
{
char dat1[20];
char lin1[80];
int     dat;
int     numb;
float   fnumb;

init_pgm(argc, argv);
printf("%s",CL);
input(0,9,10,"string prompt >>","s",dat1);
input(0,9,10,"number prompt >>","n",&numb);
input(0,9,10,"float  prompt >>","f",&fnumb);
input(0,9,10,"line   prompt >>","l",lin1);
input(0,10,10,"date   prompt >>","d",&dat);
                date_from_day(dat,stdout);

printf("%sstring stored in dat1: %s",tgoto(CM,0,11),dat1);
printf("%snumber stored in numb  %d",tgoto(CM,0,12),numb);
printf("%snumber stored in fnum: %f",tgoto(CM,0,13),fnumb);
printf("%sstring stored in lin1: %s",tgoto(CM,0,14),lin1);
printf("%sdate stored in dat:    ",tgoto(CM,0,15));
date_from_day(dat,stdout);

exit(0);
}
#endif
input(col,line,length,prompt,type,where)
int line,col,length;
char *type;
char *prompt;
int where;
/*****************************************************************
*                                                                *
*  col - column to begin printing prompt at                      *
*              left edge of screen is column 0.                  *
*  line - line number to locate Prompt on                        *
*              top line of screen is line 0.                     *
*  length - max # of characters to get from input                *
*  prompt - text of prompt.                                      *
*  type   - type of data to retrieve                             *
*              s - single string                                 *
```

Continued

```
 53  *               n - number                                      *
 54  *               l - single line (cr) terminated                 *
 55  *               d - date mm/dd/yy format input                  *
 56  *                       stored in julian integer                *

File  INPUT.C              Monday, April 14, 1986   18:34:39

 57  *  where - variable to store input into                         *
 58  *                                                               *
 59  *                                                               *
 60  *  SAMPLE:                                                      *
 61  *                                                               *
 62  *  input(0,9,"string prompt >>","s",datl);                      *
 63  *  input(0,9,"number prompt >>","n",&numb);                     *
 64  *  input(0,9,"line   prompt >>","l",linl);                      *
 65  *  input(0,9,"date   prompt >>","d",&dat);                      *
 66  ****************************************************************/
 67  {
 68  int offset,mo,day,yr,i;
 69  char duMMyC1];
 70
 71  printf("%s",tgoto(CM,col,line));
 72  printf("%s ",prompt);
 73  if (*type=='d'){
 74          printf("mm/dd/yy");
 75          printf("%s",tgoto(CM,col+strlen(prompt)+1,line));
 76          }
 77          else{
 78          printf("[");
 79          for (i=1;i<=length;i++)
 80                  printf(".");
 81          printf("]%s",tgoto(CM,col+strlen(prompt)+2,line));
 82          }
 83  ans_length=getline(ans,50);
 84  ans[length]='\0';
 85
 86  switch(*type){
 87          case 's':       /* string */
 88                  sscanf(ans,"%s",where);
 89                  break;
 90          case 'n':       /* number */
 91                  sscanf(ans,"%d",where);
 92                  break;
 93          case 'f':       /* floating point number */
 94                  sscanf(ans,"%f",where);
 95                  break;
 96          case 'l':       /* line */
 97                  strcpy(where,ans);
 98                  break;
 99          case 'd':       /* date */
100                  sscanf(ans,"%2d%c%2d%c%2d",&mo,duMMy,&day,duMMy,&yr);
101                  to_julian(where,mo,day,yr);
102                  printf("%s",tgoto(CM,col+strlen(prompt)+1,line));
103                  break;
104          }
105  }
106
107 /*****************************************************************/
108
109
110 date_from_day(inp_jday,dev)
111 int inp_jday;
112 int dev;
113 {
114 long mo,yr,dy;
115 long temp,temp1;
116 long jday;
117
118
119 jday=(long)inp_jday+735785L;
120
121 dy = jday;
122 yr = (dy-122.1)/365.25;
123 temp = 365.25*yr;
124 mo = (dy-temp)/30.6001;
125 temp1 = 30.6001*mo;
126 dy=dy-temp-temp1;
127
128 if (mo<14)
129         mo -= 1;
130 else
131         mo-= 13;
132
133 if (mo == 1 || mo == 2)
134         yr += 1;
135 yr -= 1900;
136
137         fprintf(dev,"%2.0d/%2.0d/%2.0d",mo,dy,yr);
138 }
139 /*****************************************************************/
140 to_julian(where,mo,dy,yr)
141 int mo,dy,yr,*where;
142 {
143 long jday=0;
144 if (yr==0){
145         yr=80;
146         mo=1;
147         dy=1;
148         }
149 yr+=1900;
150
151 /* routine should be good from 1935 to 2099 appx */
152 /* the base year is 2020                         */
153 if (mo>2)
154         mo += 1;
155 else{
156         if (mo==1 || mo == 2)
157         {
158         yr -= 1;
159         mo += 13;
160         }
161 }
162 jday=(double)(365.25*yr);
163 jday+=(double)(30.6001*mo);
164 jday+=(double)(dy-735785L);
165
166 if(jday>32766 || jday < -32766){
167         jday=0;
168         syserr("To Julian: Value out of bounds");
169         }
170 *(where)=((int)jday);
171 }
172 #ifdef TESTINPUT
173 #include "init.c";
174 #endif
```

OUTPUT...

Themes and Such

George Collier's letter (see INPUT) raised a number of worthy points. I too was pleased that we could put together an issue of I/O NEWS that had a theme. What made it possible was the cooperation and efforts of those who contributed the articles. We would like to do it again, and ask your help.

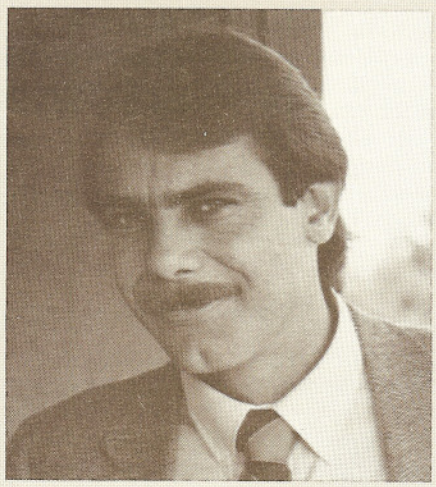
You probably wouldn't be reading this now if you didn't use a microcomputer. Most likely it's a Cromemco. No doubt you've had your share of experiences, good and bad, along the way. Perhaps you program. Or it may be that you are strictly an applications end-user. In either case, you've made your way along the learning curve, and therefore have something to share. Be it a program you've designed, a problem you've encountered and solved, or something as simple as observations and impressions of your day-to-day computer use. Maybe there's some package you use (WordStar, Writemaster, dBASE, Informix, Multiplan, etc., etc.) that you know inside out, and have put it to use in ways others may not have considered. What may appear trivial to you could be a revelation to others.

You could help by sharing that knowledge with others. The medium could be the pages of I/O NEWS. All it would take is your wordprocessor and a little time. You may even discover a latent talent for writing. It might even be fun. And if that's not enough incentive, if your article is accepted for publication, you will in turn receive a free one-year renewal of your IACU membership.

With a large base of contributed editorial materials from which to draw, it would be much easier to present such theme issues. So pick a topic and write about it (and get it to us before a deadline — see page 21 for I/O NEWS Production Schedule)!

IACU Software Resource Guide

Mr. Collier's point about issuing updates to the SRG is well taken. We anticipate that the number of references will grow in time, and so it makes sense to issue updates containing just those references gained since the last edition. Although we originally considered that possibility, for expediency in releasing the present information it was decided



Bill Jaenicke

to stick with the original format. Ultimately, we hope to go to a binder format with tabbed dividers into which updated indexes and materials could be inserted, and to offer such an update service.

Of course, all of this presumes that we will be kept constantly up-to-date on old and new software developments. I was a bit disappointed in the relatively few number of **Software Resource Surveys** that have been completed and returned so far. I urge you to take the time to photocopy the enclosed Software Resource Survey (page 37), fill it in, and return it to us. It's the only way to quickly expand the contents of the SRG.

Leverage in Numbers

Those of us who use Cromix are aware of all the great features it offers, its power and speed, and yes, of its helpful and forgiving user interface. If only the rest of the world knew, particularly software houses. And so, when one or two voices try to be heard above the din of PC and Mac users demanding WHATZIT for their machines, it doesn't come as too great a surprise when WHATZIT is never ported for Cromix.

The story would be different if, instead of one or two voices there were one or two hundred, or better yet, one or two thousand. Developers might be a little more inclined to listen. I've heard tell that there are something like 40,000 Cromix systems scattered around the world. That's a sizeable number. Banded together, even part of that would make for a loud voice.

The IACU is in a perfect position to coordinate and direct such mass requests. It would be much easier to approach software developers knowing that there was a sizeable market of Cromix users waiting in the wings. So write directly to the developer, and at the same time write a letter to the IACU. We have the tool to compile the requests (called a CS-100). And with that, we'll have some leverage.

What! No SOFT TIPS?

You may have noticed an omission from this issue's table of contents. There is no SOFT TIPS column. For a host of reasons, but mainly due to rigorous demands of his company, **Computer Specialists & Associates**, Norman Vadnais will no longer be able to edit the column on a regular basis. However, we look forward to Norm's continued contributions in the form of special articles, and wish to thank him for the fine job he did as editor.

Personally, I think there is a need for a regularly appearing column, like SOFT TIPS, which is not programmer-oriented. In short, a place to discuss the applications packages that we all use. But we need someone to put together such a column. That involves sharing your own experiences as well as those that other users will relate to you. Geographic proximity is not a prerequisite: Tom Beer (editor of C-10 ENCOUNTERS), down under in Australia, will attest to that. If you're interested in taking over the SOFT TIPS column, get in touch with me for the specifics.

Looking forward to hearing from more of you,

Bill Jaenicke
Editor

CS-200

Continued from front cover

One other big improvement in the CS-200 is in the area of over- current protection. While the CS-2 used fuses for this function, the CS-200 uses reset-table circuit breakers. This is a big plus for anyone who has ever had a fuse blow with no spare on hand!

The CS-200 is the latest edition of a product line with a proud heritage. Cromemco developed the original Z-2 computer in 1976. At the time, it was by far the most powerful micro on the market. In 1977 Cromemco became the first company in the world to include floppy disk drives as an integral part of a computer system by mounting floppy drives in the Z-2 front panel. This became the model Z2-D. Then, in 1979, Cromemco shortened the 21-slot card cage to 12-slots in order to make room for a 10-megabyte IMI hard disk in what was known as the Z2-H. Along the way the Z2-D got a name change to the Cromemco System 2 or CS-2.

When 5-inch hard disks first became available, in 1982, Cromemco then introduced what was called the CS-2H. The 5-inch hard disk was sufficiently smaller than the 8-inch hard disk of the Z2-H that the full 21-slot board capacity could be restored to the system. The capacity of CS-2H hard disk in 1982 was 5 megabytes.

Today the System 200 continues in the tradition established by Cromemco with the Z-2 computer a decade ago. It is a system that is rugged, powerful, and expandable far beyond other machines in the market. The CS-200 is available with either a 50 megabyte or 150 megabyte hard disk and can accommodate up to 16 megabytes of memory. The system fits in a standard 19-inch instrumentation rack and has easy access for custom configurations. A summary of CS-200 capabilities is shown in Table 2.

With the introduction of the CS-200 Cromemco has completed the process of upgrading all its systems to the modern age of 32 bit processors and UNIX-compatibility. This process began with the transition from the System 1 and System 3 to the System 100 and System 300 respectively, and with the introduction of the System 400 last year. The CS-100, CS-300, and CS-400 now have a new, proud sibling. CD

TABLE 1: CS-2/CS-200 COMPARISON

	CS-2	CS-200
Power Supply Output Capacity	300 Volt-Amps	450 Volt-Amps
Cooling Airflow	47 CFM	81 CFM
Number of Fans/Blowers	1	3
Number of I/O Connectors	10	20
Overload Protections	Fuses	Circuit Breakers

TABLE 2

MODEL	CS200H50X10	CS200H50X20E	CS200H150X10	CS200H150X20E
Processor	68010			
RAM	1Mb	2Mb	1Mb	2Mb
Error-Correcting Memory	No	Yes	No	Yes
ROM Firmware	RDOS diagnostics			
Serial Interface	RS-232 or current loop			
Floppy Disk Storage	2-390Kb			
Hard Disk Storage	50Mb	50Mb	140Mb	140Mb
Boards Supplies	XPU XMM STDC 64FDC 1024KZ	XPU XMM STDC 64FDC 2048MSU MCU	XPU XMM STDC 64FDC 1024KZ	XPU XMM STDC 64FDC 2048MSU MCU
Board Capacity	21 Boards			
Open Board Slots	16	15	16	15
Bus	IEEE-696			
Operating Systems Installed†	UNIX System V and Cromix-Plus			
Power	Operated from 100/115/130/220/240/260 volts, 50/60-cycle			
Power Consumption**	500 watts			
Power Supply	+ 8 volts @ 30A, + 16 volts @ 10A, – 16 volts @ 5A			
Dimensions	12-1/4"H × 19"W × 20-3/4"D (31.1 × 48.2 × 52.7 cm)			
Weight	65 lb.			
Mounting	For Rack Mounting			
Operating Environment	10-40° centigrade			

*Unformatted Capacity
**Maximum power consumption based on all slots being used
†Other models are available with Cromix-Plus only.

APPGEN

Continued from front cover

1) Produce complex business applications without programming or generating code.

2) Develop business applications 10 to 20 times faster than with COBOL.

3) Modify APPGEN-based applications 20 to 50 times faster.

4) Automatically document any application and update the documentation as changes are made.

5) Operate these applications without a significant performance penalty in speed or memory.

6) Run APPGEN applications, without recompilation, on over 50 different machines.

APPGEN, Claims checked out

My plan of action was to evaluate these claims and then evaluate APPGEN as a whole. In order to do this, I chose to write an application in both 'C' and in APPGEN. The application was a short general ledger system with a complete chart of accounts and various output reports.

1) I found that in APPGEN, I was in fact able to produce a complete application without generating any code. I encountered few real problems that could not be solved by judicious planning. In a few cases, I had to rethink my program to fit APPGEN, rather than the other way around. Instead of coding, APPGEN goes through a series of menus and screen prompts. It is much easier to learn to create and modify applications in APPGEN that it is to learn all that is necessary to code a program.

2) Development of my business application was about four times faster using APPGEN than writing it in 'C'. I suspect that it was not higher because many of my 'C' modules had already been written and debugged previously. These modules had direct application in terms of screen layouts, menu setups and report outputs. A great deal of my time in 'C', was spent compiling small changes in my programs. In APPGEN, long compiles for small changes were not needed. In addition, I made less errors in APPGEN than in 'C'. Leaving off a semi-colon or typing ' = ' instead of ' = = ' reduced my productivity.

3) As a programmer, I have found that most of my client's money is spent on modifying programs already written. This is where APPGEN truly excels. I was able to make afterthought changes to my APPGEN program approximately 32 times faster than I was able to do it in my 'C' programs.

The most common and the most significant changes made by my clients has always been changing the size or the number of fields for data entry and storage. In 'C', this requires 1) recompiling all the programs which use the data that needs to be changed, 2) changing the programs to reflect the new fields, and 3) writing a program to read the old data and rewrite it in the proper format. In BASIC, it is even worse. BASIC usually requires a rewrite of every program that contains the new field, and of every program that doesn't use the field but reads a record which contains that field. In APPGEN, simply make the change, place it in the appropriate screens, and it is modified everywhere necessary.

4) I didn't really document my 'C' pro-

gram, a major failing of mine as a programmer. It's something I and many programmers simply don't do. (Just like many end-users should back up regularly but don't). In APPGEN, each change generally prompted for additional information. The additional information is both the on-line help files and the manual documentation. While I did not extensively comment, I did have a complete series of on-line help items and I was able to print out a manual regarding my application. This is part of APPGEN and is a very nice feature. The bulk of documentation is already written as you create the application and requires little modification to make it worthwhile. Both end-user and Programmer documentation is created. The documentation follows the information typed in for the on-line manual. Because of that, it seems that the written documentation isn't really needed for end-users because anything they need to know can be gotten on-line when necessary.

5) Running the 'C' application I had written and comparing it against the APPGEN application showed no significant differences in either speed or memory. While my database was fairly small, because of the method APPGEN uses to obtain records, I suspect that record retrieval is fast even with a large database.

6) While I was unable to test my APPGEN application on other machines, I suspect that I would have little or no trouble because APPGEN is set up through the use of parameter files, rather than actual code.

In all, the claims made by Software Express check out. These are significant claims because they address the problems that are most important to both programmers and end-users — TIME and MONEY.

What is a fourth generation language?

First generation languages are Machine languages, essentially strings of 0's and 1's. My business associate loves to tell me of the 'old' days (1976) when he programmed his micro by flipping switches on and off.

A step above machine language are the Assembly languages — the second generation. Each assembly language instruction was essentially a mnemonic for a machine instruction.

Common examples of third generation languages include COBOL, FORTRAN, and BASIC. They are often called procedural languages. Many procedural languages (and even assemblers) have the capability of being written in modules and compiled separately. This capability is called modular coding. I can write a routine, place it into a common library and have this routine available to all my future programs simply by linking my program with my library routines. This is a valuable aid, because once a module is written and debugged, it need not be written again, but can be used over and over.

Fourth Generation Languages are generally broken into three categories: DBMS's, Procedural Languages, and Non-Procedural Application Generators. 4GL's are often referred to as Application Generators. They are powerful programming tools that help users to write applications, often times without the need for coding. Typically, these languages are designed for specific groups of applications. APPGEN is well suited to accounting applications. Other fourth generation languages may be just the other way around.

Fifth generation languages may include natural language, voice-to-text, artificial intelligence, and capabilities we have not yet thought of.

APPGEN as a whole

Software Express, the creators of APPGEN, use their own product. Their in-house accounting and record keeping is done using APPGEN applications. In fact, APPGEN itself is largely written in APPGEN.

APPGEN can be broken down into several different modules:

Menu Generator. When designing applications, the most frequent starting point is creating the menus that will interface the user with the system. Through a series of prompts, the user creates the menu format, menu headings and the modules and/or programs that will run from a specific selection.

Screen Generator. The most common functions found in programs that must interact with data are: ADD, DELETE, UPDATE and REVIEW a record. This is usually done with a data entry screen. APPGEN allows multiple screens and unrestricted prompt and response placement. In each mode (add, delete, update, review), conditions can be set. For example, only a certain range of values can be entered, or data can be changed only if a certain record exists, etc. Creation of the screens is done through a series of prompts. I found this to be tedious. Software Express indicates that they are working on an interactive screen creator which will eliminate many of the prompts and make screen creation a much easier task. Conditions and commands can be entered for each data entry item. Whenever I had more than three or four conditions I wished to place on a specific entry, there was insufficient room to enter all the commands. There are ways around this problem, but it occurred frequently enough for me to mention here.

Report Generator. Output is the purpose of most of our Application Programs. Output would include reports, checks, special forms, invoices, purchase orders and much more. The reports were easy to create. In creating them, the user can select on any field, can sort on any field, can produce totals, and has complete control over the output.

Data Base Management. APPGEN has variable length fields and records. This allows a tremendous amount of savings in terms of disk space. The database incorporates automatic multi-user file/record locking.

An exceptionally nice feature is multi-valued fields. A multi-valued field is one which has several values. For example, John Doe, a customer, may have a multi-valued field, invoices. The invoices would contain a list of all of the invoices in John Doe's name.

Key access is done through a hashing table. This has both advantages and disadvantages. Speed is a major advantage. Once a key is chosen, the record can usually be found with only one disk read. In a B-tree or ISAM setup, finding the key in question may take three or four reads. The disadvantages are: 1) Requires each key to be unique, 2) The key entry must be exact, "closest key recognition" is not possible, and 3) it is not easy to select and retrieve information by a range of keys. Hash tables also need to be increased in size when there are too many keys in the table. This is easily done and may be an inconvenience but is not really a disadvantage.

In accounting applications, there is usually a customer number, a vendor number, or an employee number and so finding a closest key is not a problem. However, in an application where it is necessary to find a person by name, and it starts with "Dia????", this might not be helpful. In fairness, I must mention that APPGEN does have a table lookup feature which will find all occurrences of a name where the first three

characters of each word are the same as the users selection.

Because of the way the database is structured, APPGEN has a very impressive capability. A user can modify a field (or several fields) in an application, and after completing those changes, continue using the application WITHOUT having to do any manipulation of the database. Talk about on-line changes! This feature alone can save thousands of dollars in programmer maintenance time.

Posting Integrator. One of the most important functions in accounting applications is Posting. APPGEN will integrate multiple files, update multiple fields and records in multiple files across applications boundaries, add new fields, reset accumulators and delete records.

End-users

Most of what has been discussed applies to software developers and companies that choose to maintain their own programs. Many end-users will not be able to justify, however, the cost of the APPGEN development package.

The vast majority of users will acquire the Run-time APPGEN and not the development package. In addition, they will acquire the application software and have their dealer, or a software consultant like myself, customize the application to their specific needs.

Customization of programs already written will be substantially faster and easier than writing new programs from scratch. A consultant can complete the modifications on his own equipment, and transfer the files via modem to the end-user machine. Quick modifications can be made by calling up the end-user machine on the modem and making the modifications.

Software Express has written a number of end-user Applications. These include General Ledger, Accounts Payable, Accounts Receivable, Payroll, Inventory/Order Entry, Accountant's Client Write-Up, Job Cost, Fixed Assets, and Professional Time Reporting. I have not seen all of these packages, but promise to contact Software Express and ask if I can review them in future issues of *I/O NEWS*.

In addition to the above applications, there are over 50 other application packages that have been written by other software developers.

Conclusion

APPGEN is an excellent package, well suited to accounting applications. Most important to an end-user, is the time and money that will be saved in maintaining and modifying the software over the years.

Cromemco Q & A

Continued from front cover

them becomes confused over the intricacies of Cromemco's products.

Following are some commonly asked questions about Cromemco graphics and general processing computer products.

Q: What are the capabilities of Cromemco's graphics computer?

Cromemco does not build a graphics computer. Cromemco manufactures a series of supermicrocomputers that provide real time and non-real time graphic image storage, manipulation, and display. This distinction is important because Cromemco's systems provide functions in excess of graphics. Database capabilities, micro to mainframe communications, and office automation options are just a few of the additional functions available. So in addition to providing 2-D and 3-D image manipulation Cromemco also offers the functionality of a general processing supermicrocomputer that is not always found in dedicated graphics computers.

Q: A competitor of Cromemco's, Silicon Graphics, has announced a system that uses the 32-bit M68020 CPU. How does this affect Cromemco?

Cromemco recognizes the performance benefits of utilizing the M68020 CPU operating at 16.7MHz in graphics applications. This is one of the key reasons we have decided to announce the XXU processor board incorporating this chip. The XXU will be announced in the 2nd quarter of 1986. In addition to having the M68020 the XXU will also incorporate the M68881 math chip along with a large 16KB data/instruction cache.

Q: Does this mean that Cromemco is late to market with this new technology?

No. During the development cycle a conscious decision was made to delay announcing an M68020 based product until the chip's design had stabilized. Companies such as Sun Microsystems and Silicon Graphics who announced products around September 1985 did so with the knowledge that the processor they were selling was not guaranteed to meet the published Motorola specification. Any machine shipped in the last six months claiming to use an M68020 operating at 16.7 MHz probably utilized an XC68020 instead. The XC prefix is a Motorola designation for a part that does not meet full product specifications.

Some companies introduced systems with the M68020 as early as May 1985 (i.e.. the 2403FT from Charles River Data Systems). These products, however, utilized a slower version of the chip which operates at 12.5MHz.

Now that Motorola has finalized the design and testing on the 16.7MHz M68020 Cromemco feels confident about announcing its use. Alpha and Beta site testing is on going at select OEMs and VARs account to insure product integrity.

Q: Cromemco uses the S-100 system bus. Isn't that outdated technology?

It has been said that the S-100 bus (IEEE-696) is very old technology, and that Cromemco's implementation is non-standard. If this is so then the majority of supermicrocomputers sold today utilize old technology. Dec's MicroVax II, NCR's Tower 1632, and Silicon Graphic's Iris 2400 all utilize a 16-bit bus. The products from NCR and Silicon Graphics implement a version of Multibus (IEEE-796). Multibus is as old as the S-100 bus and also has many different versions. Standard Multibus will also not support a dedicated 32-bit processor.

On the issue of standards, even the newer buses such as VME and Multibus II suffer from differences. At last count there were three versions of the VME bus, none of which are compatible with each other.

Reliability is another issue over which there is a lot of confusion. The DIN pin/socket connectors used on the VME bus are often cited as a reason why that bus should be more reliable than Multibus or S-100 (which use PC-board edge connectors). Modern edge connectors, however, use high normal pressure and form reliable gas-tight connections which provide the same level of interconnect integrity as the DIN connectors. The S-100 also has half the number of physical pins which makes the probability of contact failure in VME worse.

An example of proven bus reliability comes from the U.S. Air Force. Cromemco systems have been deployed at over 1200 Air Force locations around the world. They have been subjected to rain, mud, and other environmental hazards no normal computer user would dream of subjecting his system to. In one instance a Cromemco computer was filled with sand and expected to function normally. Throughout this "trial under fire" Cromemco computers continue to function and provide the reliable data processing performance their users have come to expect.

Q: What is a typical image resolution that can be produced?

Cromemco can display an image 756x484x24 pixels out of a field of 1024x1024. The first number refers to the number of dots across the horizontal axis of the screen. The second refers to those along the vertical axis. The third refers to the depth of the image into the screen. The deeper an image can be made the more realistic it becomes.

Applications that can benefit from this type of resolution include video imaging, computerized publishing, graphic arts production and 3D flight simulation. Some computers offer images that are larger in length and width but are not as deep. Silicon Graphics systems have a maximum resolution of 1024x780x8. While this provides a larger flat image, it does not provide a realistic image because of the lack of picture depth. In applications such as training, realism is an added bonus because it allows the student to see what a situation will look like in real life rather than having to imagine whit it looks like based on an unrealistic modeled display.

Q: I've heard about cache memory. How does this effect a computer's performance?

Cache memory allows a computer to access and process data faster. Cache memory is essentially a block of available memory that is placed in close proximity to the CPU. It is used to store data and instructions that the CPU must get to rapidly. When cache is present the CPU does not have to make request through the system bus as often and as a result can process jobs at a much faster rate. This leads to increased system throughput and performance.

The quantity of cache memory impacts the level of productivity gains achieved. The more memory the cache has, the better the chances become that the desired data will be resident there. The cache efficiency thus goes up and so does the overall performance.

Cromemco has included 16KB of cache memory on our soon to be announced XXU processor board. This is one of the largest amount of cache memory available today on a microcomputer. Charles River pioneered cache in micros in 1982 for the 68010 processor. Since cache is of most benefit when the processor is much faster than main memory, the Charles River machine did not benefit greatly from cache. Additionally, their cache was a relatively small 4KB.

From a users standpoint this modular approach proves cheaper because functions that are not needed do not have to be purchased. Cromemco's S-Series graphics boards are a five board set that perform 2-D and 3-D graphic image digitization, storage, manipulation, and output as either an RGB signal or composite video. Many applications do not require all of these functions and thus could get by with fewer boards. Applications such as flight simulators or training workstations might not require output in composite video or multiple overlay planes with 16.7 million colors. If the five board set were condensed onto a single board, however, the user would have no choice but accept the additional functions.

The last point to be raised covers reliability and repair. The more circuits and functions that are placed on a board the higher the probability is of a failure or defect occurring. When a multifunction board develops a problem detecting it becomes a time consuming and expensive task. It even becomes expensive to swap out the bad board with a good one because of the high replacement cost.

Q: Cromemco only offers 2MB of RAM per board. Isn't this small compared to other manufacturers?

Memory capacity per board has been proposed by some observers as a measure of design quality. This is not a consistent measure, however, due to physical size differences in different types of boards. Cromemco memory boards have a maximum capacity of 2MB. This is the maximum offered by any manufacturer of S-100 based systems. Some manufacturers advertise 4MB per board. As it turns out this larger capacity is achieveable only because the board used (i.e. Multibus vs. S-100) is larger and can hold more chips. Board capacity is thus not always a meaningful measurement since having 4MB on a board twice as large costs the same as 2MB on two boards half as small.

The real measure of memory efficiency is in the number of chips per square inch of board. By this measure Cromemco has one of the highest chip packing densities in the industry.

A more practical consideration is how much memory a computer can really use. This takes into account the number of board slots, memory per board, the capability of the memory manager, and many other factors. By this measure too, Cromemco is an industry leader since they support 16MB of real memory.

Q: The Maximizer floating point co-processor board is advertised as performing at 12 MIPS. Yet a VAX 11/780 only runs at 1 MIPS. How is this possible?

The Maximizer is a user programmable co-processor that is extremely flexible for the creation of custom high performance systems. It does indeed execute 12 million instructions per second. The VAX 11/780 also has a fast floating point co-processor in it and it too executes instructions at about 12 million instructions per second. The fact that the 11/780 is rated as a 1 MIPS machine is because the CPU operates at about 1 MIPS. Linking machine performance to CPU performance is done because the CPU is involved in the bulk of the processing chores and ultimately sets the whole machine's character.

Q: If MIPS is not an appropriate measure of system performance, what is?

For calculation intensive applications such as graphics the Whetstone benchmark is an industry accepted means of comparison. A benchmark is a computer program which is run on different computers to determine the differences in performance. There are many different benchmarks that test different functions in a computer. The Whetstone suite of programs is designed to measure a system's speed in carrying out single and double precision floating point calculations. These are used in many diverse applications including graphics, statistics, mathematics and modeling/simulation.

With the soon to be announced XXU board (which includes the M68020 CPU and the M68881 math chip) Cromemco systems will perform over 1 million Whetstone calculations per second. The VAX 11/780, a dedicated minicomputer costing over four times as much as the largest Cromemco system, performs between 500,000 and 750,000 Whetstones/sec.

As with any benchmark, however, the results are not always transportable. By specially coding the Whetstone programs in the co-processor some manufacturers artificially inflate their performance ratings. Current manufacturers of supermicrocomputers claiming well over a million Whetstones/sec are either specially programming their computer or utilizing optional hardware that often drives the system price well above the published list. This is artificial because 1) the user probably will not have the expertise, the resources, or the inclination to duplicate special code, and 2) other systems in the comparison don't have the same hardware configuration. Because of this tendency to add options to a system to boost published results it is best not to judge a computer's performance solely on stated ratings. The best test still remains running a comparison involving the application program to be used.

Altos began shipping the second widely used micro with cache in the summer of 1985. Again the cache was small (4KB) and operated with the slower 12.5MHz version of the M68020.

Thus the use of cache memory on microcomputers is a relatively new technique. Out of roughly 120 vendors of 68000 based microcomputers only three are known to have had cache memory implemented for more than one year.

Every computer manufacturer who uses the M68020 processor can claim to have cache memory implemented on their computer. This is because the chip has a small amount (256 bytes) in it. Cromemco recognizes the processing speed advantages cache memory brings to a system That's why every major system element (from the CPU to Serial I/O) implements cache technology. The soon to be released XXU board alone contains 16KB of cache memory. This is 64 times more than Motorola put on their chip and twice as much as any of Cromemco's major competitors (such as Altos and DEC). Furthermore, the Cromemco cache works with data as well as instructions, and thus is even more valuable than the chip's instruction-only cache. When this sophisticated CPU cache is combined with the caches on the hard disk controller, the memory manager, the serial I/O interface, and the Ethernet board, it becomes clear that Cromemco is the leader in implementing this technology on supermicrocomputers.

Q: If the S-100 bus is considered old technology, how can I expect to get high performance graphics from a Cromemco system?

A fairly common mistake people make is to equate older technology with low performance. In many cases the older technology provides an added measure of reliability at no overall sacrifice in performance. All graphics or general processing computers have

components that are not considered state-of-the-art. this is done because 1) it is not economical to build a microcomputer with all new technology, and 2) it would take a tremendous development effort to integrate the various unrefined components.

System performance is dependent on how the individual components fit together rather than what they are by themselves. An example of this has to do with how the S-100 bus handles power distribution. The component in a system with the highest probability of failure is the power supply. It is here that Multibus and VME based systems are weakest due to their single high-stress power supply. The S-100 bus separates the power into an extremely reliable "bulk" supply and a multiplicity of low-stress on-card regulators. This is more cost-effective (since 20 15-watt regulators are cheaper than a single 300-watt regulator), and much more reliable since a single failure need not bring down the whole system. Furthermore, system-wide voltage regulation is better than in single-supply systems. This helps to eliminate transient errors caused by fluctuating loads.

Q: Cromemco products seem to emphasize using a large number of boards. Isn't this costly?

Actually this approach proves cheaper for most applications. It also provides for higher performance. By providing boards that concentrate on a single functional area it is possible to use circuitry that does not sacrifice performance for size. Cromemco's UNIX memory manager board (XMM) is a good example. The memory manager is a single board that is required for all UNIX based systems using the M68010 processor. The memory manager is a crucial part of any UNIX system, Cromemco's XMM is a state-of-the-art design modeled after memory managers used in mainframe computers; it outperforms systems which use a simple chip-set to implement memory management functions since these chips cannot achieve the requisite complexity.

Often, placing circuitry on a separate board prevents having to duplicate it on each board the customer buys. This means lower overall cost and less duplication of effort. Cromemco's implementation of error correcting (ecc) RAM, for example, utilizes one board to store the data (up to 2MB) and one board to perform memory control functions. The memory controller interfaces with up to 6 ecc memory boards and performs error detection, correction, and memory refresh chores. By separating these control functions and placing them on a separate board, Cromemco avoids duplication of this circuitry on each memory card and thus achieves cost and reliability improvements. Non-error correcting memory boards do not require a controller.

Q: Many government applications require a computer to be Tempested. Does Cromemco meet this requirement?

Cromemco in the past has left the Tempesting of their computers up to the VARs and OEMs who actually sell into the government applications market. Due to changing market conditions they have begun to reevaluate this position. Cromemco is in the process of joining the Industrial Tempest Program run by the NSA. They anticipate little problem in obtaining certification since their boxes are exceptionally well constructed and have low r.f. noise. Preliminary laboratory tests appear to support this level of confidence.

Q: Most new supermicrocomputers have diagnostic LEDs on the front panel. Why doesn't Cromemco?

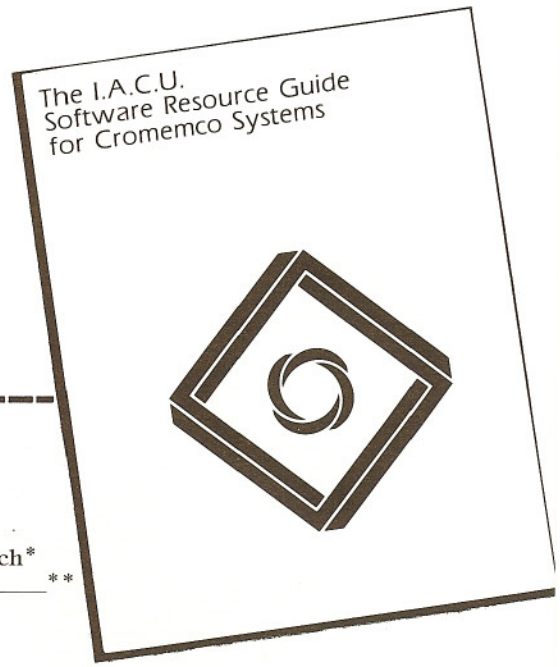
Most supermicrocomputers do not have diagnostic LEDs on the front panel. The only major manufacturer that Cromemco knows of that provides this is Silicon Graphics. Cromemco has chosen not to use LEDs because of the incompleteness of the display and because of features provided under UNIX. System diagnostic error messages in UNIX are normally routed to the Administrators Console (terminal). This is preferred by many users because the full alphanumeric capability of a terminal allows the error messages to be in English not just numbers. It also means the operator only has to concentrate on the terminal screen and not be monitoring both the screen and computer front panel at the same time.

Cromemco does have diagnostic LEDs on the XXU and other boards for use in trouble shooting at the board level. This approach provides the most diagnostic information to the operator and to the service technician as possible; printouts on the main console, disk-based system activity reports, and board level diagnostic LEDs are all provided. I/O

Having problems locating software?

Now there's help ...

The IACU Software Resource Guide for Cromemco Systems



☆ Over 180 pages

☆ Multiple indices – by product name, application group, operating system, etc.

☆ Complete product descriptions – from source language to technical support

Order your copy now! Only $50

(California residents add 6% sales tax)

The I.A.C.U.
P.O. Box 17658
Irvine, CA 92713
U.S.A.

Send __ copies @ $50.00 each*
I enclose $_____**
☐ Check ☐ Money Order
(make payable to The IACU)

Charge my ☐ MasterCard ☐ Visa
Phone orders with credit card call (714) 661-9764

Card No._____ Expires_____

Signature_____

(Please Print)
Name _____

Address _____

_____

City _____ State _____ Zip _____ Phone _____

Country_____

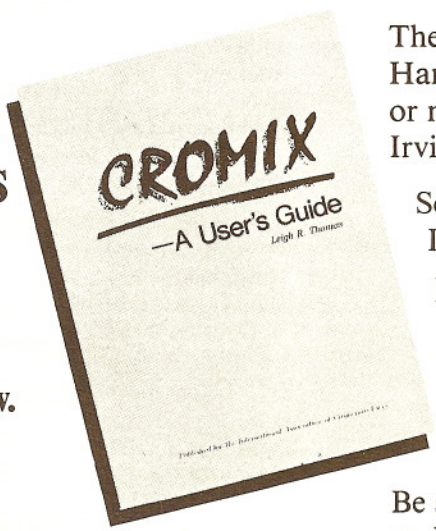
* Calif. residents add 6% sales tax
** U.S. Dollars. International checks not drawn on U.S. bank add $10

Attention CROMIX Users

The book you've been expecting is finally here.

CROMIX — A User's Guide can be ordered now.

(While They Last)



Leigh Thomas' excellent tutorial on CROMIX, reprinted in the U.S., can now be ordered directly from I/O News.

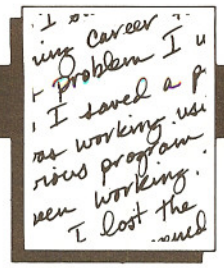
The price is $25.00 plus $4.95 Shipping & Handling per copy. For bulk orders of 10 or more, the price is $17.75 per copy F.O.B. Irvine, California. California residents add 6% sales tax

Send Check, Money Order, or Credit Card Information to:

I/O News
ATTN: CROMIX — A User's Guide
P.O. Box 17658
Irvine, CA 92713
U.S.A.

Be sure to include Membership Number and full shipping instructions with order. (Amounts noted are in U.S. dollars drawn on U.S. banks, only.)

This is a MUST BOOK for any CROMIX user.



USER NOTES

USER NOTES are useful techniques, tips, ideas and other helpful information gleaned from our member's experiences with their Cromemco systems. If you have something along these lines that you want to share, write it up and send it to I/O NEWS, c/o USER NOTES, PO BOX 17658, IRVINE, CA 92713.

As I mentioned in last issue's *OUTPUT* column, the I.A.C.U. recently acquired a CS-100 system with Cromix-Plus and UNIX System V. Quite a step up from our old workhorse — a CS-1 running under Z-80 Cromix. In the course of setting the system up, and getting to know it, I learned a few things that might be of interest to others who have recently purchased, or anticipate purchasing, a similar system.

General Impressions

In the three months of my acquaintance with the new system, I have experienced both pleasure and frustration — but hasten to add that the frustrations were few, and could easily have been prevented had I known then what I know now (and therefore wish to share with you).

The most immediately noticeable quality was the vast improvement in its overall performance. With its bigger and faster 68010 microprocessor, its STDC controlled fast hard disk, Octart powered serial I/O, and 68000 based Cromix-Plus Operating system, I had expected it to be quicker. But even that knowledge couldn't prepare me for actually seeing it in action. Programs that the old Z80 machine leisurely waded through became supercharged under Cromix-Plus and the new hardware. For example, loading SBASIC under the old system took about three seconds; under the new system loading the same program takes less than a second. This speed improvement was apparent across all system functions. Another case in point is the maintenance of the I.A.C.U. member list (the programs for maintaining the list are written in 32K Structured Basic using KSAM). When adding or updating member information, the program prints the letters of the word "SUCCESS" as each file and alternate file is updated. With the older system it took about 8 seconds for this message to appear, letter by letter. Now the "SUCCESS" message appears almost immediately. Only now, after more than three months use of the system, am I beginning to grow accustomed to this tremendous speed improvement and think of it as "normal." But from what I have heard of the soon to be announced XXU processor board, with its full 32-bit architecture and fast 16.7MHz clock, this new sense of normality may be short lived.

Less immediately apparent, but probably more important, is the extreme reliability of the new CS-100. The new system has been running continuosly and without fault. That, combined with the CTD and the knowledge that the entire system is backed up (frequently) on tape, makes for a much better night's sleep.

The System

Sometimes a "model number" is more than just a number. The model number of our system is CS100H50X10. In Cromemco's 1986 Price List there is a section on "*How to Read Cromemco Model Numbers*". Using those guidelines the above model number can be deciphered as: a Cromemco System (CS), of the 100 family (the current families are 100, 200, 300, and 400) equipped with a Hard Disk (H) with 50 megabyte (50) capacity and an XPU processor board and an XMM memory manager (X) with Unix System V and Cromix-Plus installed (other possibilites are "XC" meaning XPU with Cromix-Plus installed or "D" meaning DPU with no operating system installed), and lastly, that there is 1 Megabyte (10) of memory installed (another example: "80E" would mean 8 Megabytes of Error-correcting memory).

The system was supplied with a 64FDC, an XPU, a 1024KZ memory board, an STDC Hard Disk Controller, and an XMM memory manager. An Octart was ordered separately (although they are now standard with CS-100's).

In addition we acquired two C-5 terminals, a CTD cartridge tape unit, and a Cromemco 3032CQ dot matrix printer.

Setting It Up

Unpacking and connecting the components of the system was much more straightforward than I had anticipated. To begin with I assembled all of the various manuals and related systems documents and began reading over the pertinent materials. Although there are a number of cables which need to be connected (for the CTD and the Octart), the documentation provided clear instructions and diagrams, which made the installation of the CTD and Octart a simple matter.

The CTD runs off of the 64FDC. There is a single cable which runs from the 64FDC connector to the backplate connector (which must also be installed). The CTD has its own "piggy-back" type connector which plugs into the backplate.

Installing the Octart was a little more involved. First the board is inserted into an empty slot (near the bottom). The Octart has two serial connectors, each of which provide four serial channels. A ribbon cable connects these to the two 25 pin connectors which get installed in the back panel. The Octart cable itself is an ominous looking arrangement of four separate cables which terminate as one 25 pin connector. Two of these 4-to-1 splitter cables are necessary to utilize all eight serial channels. Five serial channels are available otherwise: four off the J1 connector and Octart cable and one from the J2 connector. I chose to use the four channels off the Octart cable for terminals and modems and the J2 channel for the serial printer.

One word of advice: have the proper tools at hand. I manag-

ed to install the cables and connectors with a crescent wrench, screwdriver, and pliers, but it would have been much easier with the appropriate socket wrenches and socket drivers. Also, when installing the internal cabling, be careful to arrange the cables so that one does not dislodge another when the back panel is secured. I disconnected the cable to the floppy disk in this fashion and suffered unnecessary anguish the first time I tried accessing the floppy drive.

The Trial Run

Having installed the Octart and connected the peripherals (terminals, CTD and printer), all that was left to do was to plug in the electrical cords and power it up. Unfortunately, my power strip had only four outlets and I needed five — one each for the CS-100, the CTD, the printer, and the two C-5 terminals. So for the time being, I didn't plug in the second C-5 (which runs off the Octart — the C-5 used as the console runs off the 64FDC serial port).

The moment of truth was at hand. I took a deep breath, turned the power switch key to right (noting the smoother action of the key mechanism), and let out a partial sigh as I heard the reassuring whine of the fans and hard disk powering up.

Having read the Technical Bulletin regarding the new RDOS 3.12 for the 64FDC, I was aware that the system could boot directly from the hard disk (either Cromix-Plus or UNIX). But, following the instructions in the System 100 manual, I inserted the Cromix-Plus boot diskette into the floppy drive, switched on the C-5 console (which emitted a beep indicating that its self-test had met with success), and, with cursor displayed on a blank screen, pressed the RETURN key a couple of times. I had expected the light on the floppy drive to come on, but it didn't. I had also expected to see the Cromix boot sequence display shown in the documentation, but didn't. Instead, I was confronted with the UNIX "#" prompt. The system had automatically booted UNIX from the hard disk, which was a relief (the hardware was working) and unsettling (it hadn't done what I thought it would do).

A quick scan through the *Introduction to UNIX System V for Cromemco Systems* provided the proper procedure for shutdown from this level of UNIX (single-user mode) — repeated sync commands to flush the buffers followed by a system reset or power down. At this point I took the easy way out and called Rich Quinn at QUINN TEAM. I learned that Cromemco had recently begun configuring their UNIX systems such that they booted directly to the UNIX partition on the hard disk, and that this change had not yet been reflected in the documentation. To boot Cromix under this configuration it is necessary to abort the automatic booting upon reset using CONTROL-C. Then, from the RDOS prompt you can boot directly to the Cromix-Plus partition (on STD0) using the command: bst0 (to boot off the floppy the command would be ba).

To get the system to boot to the Cromix-Plus partition upon power up or reset, all that is necessary is to issue the command wboot std31 (while running under Cromix-Plus). This writes the Cromix-Plus boot track to the disk partition that describes the entire hard disk (std31). Having accomplished that, I reset the system and gave it a try. Sure enough, there was the memory test described in the documentation and the familiar Cromix logon sequence (date, time, Login).

Now the real education began, and I am indebted to Rich Quinn and Mike Hazen of Quinn Team and Michael Petersen of Accountability Systems for all of the assistance they provided. It quickly became apparent that there was a good deal of configuring that needed to be done before I could put the system to full use. For one, the operating system did not yet know of the existence of the Octart or the CTD so I couldn't yet try the printer, CTD, or other terminal. Secondly, I learned that the disk partitioning done at the factory assumed that UNIX would be the predominantly used operating system and therefore had allocated the majority of disk space as UNIX partitions. Since I planned on using Cromix-Plus for the bulk of data processing (at least in the beginning), but still wanted UNIX available for educational purposes, it would be necessary to redefine the existing disk partitions and assign more hard disk storage to Cromix. This meant that the entire drive would have to be reinitialized and the Cromix-Plus and UNIX operating systems reinstalled (ughh!!). Installing Cromix-Plus from the five factory floppy disks wouldn't be bad, but the prospect of tediously reconstructing the unfamiliar UNIX operating system from its set of twenty-nine floppies was not very appealing.

But that's what cartridge tape drives are all about. So the game plan became clear. I would have to "gen" a version of Cromix that supported the CTD and Octart, and using that version, copy the Cromix and UNIX partitions on the hard disk to tape, initialize the hard disk and reassign the disk partitioning, and then reinstall Cromix and UNIX from tape. No problem.

Configuring Cromix-Plus: Sysdef and Crogen

One of the facilities that sets Cromix-Plus apart from its earlier counterparts is the manner in which it handles the generation of tailored versions of itself. The older Z-80 Cromix had this facility: the crogen utility, which through a series of prompts queried the user as to the desired configuration. This was all well and good, and much more user-oriented than the awkward mechanism provided in UNIX. Cromix-Plus takes it a step further with its incorporation of a sysdef file (a part of the original sysdef file is shown in Fig. 1).

FIGURE 1.

```
%       Cromix 3D System Generation file
%
%       Sep 20, 1985

% Device driver names should be entered on appropriate row. A current
% list of devices supported and their driver names can be found at
% the end of this file. Each driver can have a number of integer
% arguments. Those arguments, if any, should follow the driver name.
% The arguments must be separated by white space. The number of arguments
% and their meaning depend on the particular driver. See description
% at the end for the arguments a driver might require.

% System memory size:

        maxmem  2               % Amount of supported memory expressed
                                % in 256K units.

% Character devices:

        CDEV    01      tty 0                   % FDC terminal only
        CDEV    02                              % Suggested qtty
        CDEV    03      sysdev                  % System driver (required)
        CDEV    04      timer                   % Timer driver (required)
        CDEV    05                              % Suggested lpt
        CDEV    06                              % Suggested typ
        CDEV    07                              % Suggested slpt
        CDEV    08                              % Not used
        CDEV    09                              % Suggested qslpt
        CDEV    10                              % Suggested ffp
        CDEV    11                              % Suggested tape
        CDEV    12                              % Suggested cnet

% Block devices:

        BDEV    01      cflop                   % Cromemco floppy driver
        BDEV    02                              % Suggested uflop
        BDEV    03      allmem                  % Amem driver (required)
        BDEV    04                              % Suggested tflop
        BDEV    05                              % Suggested ramdsk
        BDEV    06      stdc                    % STDC driver
        BDEV    07      smd 0                   % Removable part of SMD 0
        BDEV    08      hd                      % IMI hard disk

% Primitive terminal device:

        RAW     raw_fdc                 % FDC primitive terminal driver

% Root device:

        ROOT    none            % ROOT none      (Means: Ask the operator)
                                % ROOT boot      (Means: Same as boot disk)
                                % ROOT 6 0       (Means: Use device 6:0)

% Automatic login name:

        LOGIN   system          % LOGIN system     (For Boot System)
                                % LOGIN            (For no auto login)


% Customized logon message:

        LOGMSG  Boot System                     % Any message can be here


% Default access:

        ACCESS  rewa.re.re      % Files created will have this access
                                % unless it is changed here
```

Continued

```
%                       SYSTEM PARAMETERS
%
% NOTE: Be sure you are aware of the ramifications of altering
%       these values prior to changing them.  See Cromix-Plus User's
%       Reference Manual 023-5013 for details.

        bufcnt  30      % Number of memory resident data blocks

        inocnt  30      % Number of memory resident inodes

        filcnt  80      % Number of files which can be opened simultaneously

        usrcnt  24      % Number of process tables

        logcnt  24      % Number of users who can be logged in at the same time

        mntcnt  8       % Number of devices that can be mounted at any time

        lckcnt  16      % Number of locks that can be installed

        freecnt 2048    % Number of bytes in the system memory pool

        END
```

Under Cromix-Plus, the sysdef file is used as the template for specifying the configuration of a new version of the operating system. It is a standard text file which can be edited with screen or ce (it also contains extensive documentation about itself and how to use it). The system parameters are system memory size, character devices employed, block devices used, specification of the root device, automatic login name, customized logon message, and default file access attributes. The sysdef file serves as input to crogen (rather than prompts), which generates the desired version of Cromix. The online help files for crogen and sysdef are extensive.

This method of system generation has many benefits. Besides being easy to modify, you have a text document describing the configuration of the system. For service technicians or system administrators it makes it an easy matter to determine the parameters of the system configuration. Another nice feature is that the sysdef file does not have to be called "sysdef." Nor does the generated version of Cromix need to be called "cromix.sys." When using Crogen you can pass both the name of the sysdef file and the desired name of the tailored version of Cromix as parameters.

I created a custom system definition file (in the /gen directory) by using ce to create a file called iosysdef, and then by reading into that file the original /gen/sysdef file. To tailor a new version of Cromix-Plus to my particular hardware configuration I made the following changes in the iosysdef file:

1) Changed maxmem from 2 to 4 (4 units of 245K = 1024K of memory available)
2) Added qtty1 to the CDEV 02 line int the Character devices section.
3) Added qslpt 1 to the CDEV 09 line in the Character devices section.
4) Added tflop 4 to the BDEV 04 line in the Block devices section.
5) Added ramdsk to the BDEV 05 line in the Block devices section (necessary if you intend to use Ram disk).
6) Removed the smd 0 specification from the BDEV 07 line (removable SMD driver not required).
7) Removed hd from the BDEV 08 line (no IMI hard disk being used).
8) Changed the ROOT specification from none to 6 0 (boot off STD0).
9) Changed the automatic LOGIN name from system to no name (blanks for no automatic login).
10) Customized the logon message to read "Welcome to the IACU System 100."
11) Changed the default file access to rewa.rewa.re so that members of a group would have full access to files created by other members of the same group.

Saving these changes in my new iosysdef file, I then proceeded to complete reconfiguration:

1) Using screen (with the -n option) I modified the /etc/ttys file, changing the designation of tty1 from dumb to C-05 and turning qtty2 "on" (by changing the 0 to 1). For qtty2 I left the baud rate automatic, but changed the designation of the terminal from dumb to C-05. This terminal description aspect of the /etc/ttys file for use with termcaps is another distinction between Cromix-Plus and its earlier counterparts.

2) Before I could use the printer it was necessary to first create the appropriate device file and link it to the prt device file. Recall that I had hooked the printer up to the J2 connector on the Octart. This meant that I would be using the 5th serial channel (the first four channels are on J1, to which I had connected the 4-to-1 cable and intended to use for additional terminals and the modem). Therefore, I would need a qslpt5 entry in the /dev file. The command makdev qslpt5 c 9 4 accomplished that (the c indicates a character device; the 9 is the major device number corresponding to the CDEV 09 qslpt 1 entry in iosysdef; the 4 is minor device number, and because they start with 0, corresponds to the 5th qslpt). Having created the qslpt5 device, I then linked it to the system printer device file using the command maklink -f /dev/qslpt5 /dev/prt.

3) Since I would be using an Octart in the system, I had to edit the /etc/iostartup.cmd so that it would download the Octart drivers upon system boot. This simply took removing the % (comment sign) from the line in the file which read /etc/octload /etc/octart.iop io1. The /etc/iostartup.cmd file has all of the commands which could potentially be used preceded by a % sign, i.e., as comments in the command file. Activating the command is accomplished by deleting the % from the line.

4) I added a line to the /etc/startup.cmd: startsys. Startsys.cmd is a command file in which I placed the various terminal and printer mode settings I desired to be set when booting the system, e.g., mode prt bm 0 baud 9600.

Having made all of the above noted modifications, I was ready to generate my new functional version of Cromix. The command crogen cromix iosysdef, given while in the /gen directory, accomplished this. After completion, I moved the newly generated cromix.sys file from /gen to the root (having first renamed the original cromix.sys file to cromix.old).

Now, another moment of truth was at hand. I ran shutdown and reset the system. The memory check displayed on the screen, followed by the Cromix logon sequence. I logged on. Apparently all had gone well, but the true test was whether or not the printer would respond. I turned the printer on, loaded it with paper, put it online and typed in the command echo "hello" V /dev/prt. The printer responded, and "hello" appeared on the paper. Relief turned to elation. It worked!

Partitioning the Hard Disk

Now I was equipped to set about repartitioning the hard disk for use under Cromix. But before going into the details, I'll give some general information regarding STDC drives.

One of the great features of STDC hard disks is that the physical drive can be partitioned into a number of logically separate and distinct disk drives, labelled std0 through std30. Each of these partitions is capable of being a root device, or a mountable disk. Std31 is special, and designates the disk drive as a whole.

Our system was supplied with the following standard partition for our type of 50 meg drive:

STD0 = Cromix-Plus root partition
STD1 = UNIX swap area (memory to disk)
STD2 = UNIX root
STD3 = UNIX data partition (to be mounted under UNIX)
STD4 = FTAR swap area (file transfer between Cromix and UNIX).

Cromix-Plus has two utilities for examining the configuration of a drive: diskinfo (new) and mode (enhanced to work with block devices). Both of these utilities are thoroughly documented in the online help. Diskinfo displays the physical and logical configuration of a disk drive in terms of number of heads, cylinders, etc. It also presents the partition table and alternate track table for the drive (see Fig. 2).

FIGURE 2.

```
Device /dev/std31 6:31  STDC Hard Disk  Jan-27-1986 12:12:10

Disk parameters
        Number of heads..................... = 7
        Number of cylinders................. = 714
        Number of alternate tracks.......... = 56
        Start of write precompensation...... = 256
        Location of alternate tracks........ = 353

        Number of sectors per track......... = 20
        Bytes per sector.................... = 512
        Location of alternate track table .. = 9792
        Location of partition table......... = 9728
        Starting cylinder of disk........... = 0
        Disk label.......................... = CSTD

Partition table
   43  103  391  679

Alternate track table
  # bad track alt track      # bad track alt track      # bad track alt track
    hd   cyl  hd   cyl         hd   cyl  hd   cyl         hd   cyl  hd   cyl
 0   0    36   0   353     1   0   132   1   353     2   0   169   2   353
 3   0   242   4   353     4   1   283   6   353     5   4   501   1   354
 6   6   223   3   353     7   6   248   5   353     8   6   467   0   354
```

Use the command diskinfo std31 to get the information about the entire drive. To get a hardcopy listing of this information (which is a good thing to have on hand) you can pipe the output via the spool utility to the printer using the command: diskinfo std31 | spool.

With the information provided by diskinfo you can determine the size of the various partitions. The relationship between the number of bytes in a partition is governed by the number of read/write heads and cylinders: Partition size = (No. of heads * 10) * (No. of cylinders).

For example: examining the Disk Parameters section shown in Figure 2 reveals that the drive has seven (7) heads. The partition table shows that cylinders 0 through 42 are assigned to STD0 (a total of 43 cylinders). Applying the formula yields (7*10)*43 = 3010K. Likewise, STD1 is comprised of cylinders 43 through 103 (60 cylinders) and is 70*60 = 4200K large. In this fashion the sizes of the STD2, STD3, and STD4 can be determined.

There is an easier way to determine the sizes of the disk partitions using the mode utility. For example, among other items such as the drive's RPM, format (CDOS/CROMIX), and version of inithard used, the command mode std0 reveals that std0 has 43 cylinders and is 3010K. Doing mode for each partition yielded the same partition sizes as could be computed using the formula above:

Drive	# of Cylinders	Size in KBytes	Purpose
std0	43	3,010	Cromix Root
std1	60	4,200	UNIX swap space
std2	288	20,160	UNIX root
std3	288	20,160	UNIX data
std4	27	1,890	FTAR swap area

As I've mentioned, this wasn't the configuration I wanted. Although I did want UNIX available, I intended to use Cromix-Plus for data processing, and therefore needed to allocate some of the UNIX data disk space to the Cromix root partition (for application software added) and to a Cromix data partition (which I would first have to define).

The desired configuration is represented in Figure 3. The UNIX root and UNIX swap space partitions were retained as is for use with UNIX, as was the FTAR swap area (the Cromix File Structure Utility (CFSU), which appeared in *I/O NEWS, Vol. V, No. 1*, relieves this requirement of a commonly shared partition used in the transfer of files between UNIX and Cromix partitions). Basically, I took away from the UNIX data partition and gave to the Cromix root and data partitions.

Figure 3.

Drive	Cylinders	# Cylinders	Size	Use
std0	0—84	85	5950K	Cromix + Root
std1	85—144	60	4260K	UNIX Swap
std2	145—432	288	20160K	UNIX Root
std3	433—607	175	12250K	Cromix + Data
std4	608—678	71	4970K	UNIX Data
std5	679—705	27	1890K	FTAR Swap

Backing Up

I now had all the tools necessary to copy from disk to tape, and knew how I wanted the hard disk partitioned. As it stood, all I really needed to copy were the UNIX root and Cromix root. And since I intended to make frequent use of the Cromix data partition, I wanted a tape for that also.

Backing up the entire system would prove a true test of the CTD. I knew that there would come the time when the hard disk would be blank — all of the information that it had once held would now reside only on those seemingly fragile ribbons of magnetic media. And it would have to be perfect, and restorable. Visions of audio cassettes being chewed to pieces by tape decks flashed through my mind, accompanied with the nightmarish thought of jockeying all of those system floppies and unifying the intricate software scattered among them.

Needless to say, I took every precaution I could in making those first tape backups. When I purchased the tapes (which aren't cheap at $30 a throw) I made sure that the tapes had not been manufactured between the 15th and 37th weeks of 1985 (see *Tec Tips*, in *I/O NEWS, Vol. V, No. 2* for how and why). I also retensioned them a couple of times before intitializing, as was recommended in the *Tec Tips* noted above.

Another enhancement to Cromix-Plus became apparent. Where the older Z-80 version had only one utility, init, for initializing floppies and hard disks, Cromix-Plus is endowed with three separate utilities: initflop for initializing floppy disks; inithard for initializing hard disks; and inittape for intitializing cartridge tapes.

The online help file provided me with all the information necessary to use the inittape utility. The program prompts are straightforward enough. I used all of the default responses. The one thing that took me by surprise was the length of time it took to complete the initialization and verfication process — nearly half an hour! Later, I only used one verification pass (the default is three), and thereby shortened the time.

I had learned that I would be using two different Cromix utilities for backing up to the tape: ftar for the Cromix partition and rcopy for the UNIX partition. Again, the online help provided extensive documentation on both of these utilities.

The Cromix root partition was copied to tape using the following set of commands:

```
# d / (go to root = std0)
# ftar -cv /dev/ftcd
# ftar -yv /dev/ftcd
```

The first ftar command creates a new archive on the tape (-c) and lists the files as they are copied (-v). The period specifies that the current directory is to be backed up — file names on the tape will be preceded with "./", e.g., ./cromix.sys or ./bin/ftar.bin. Thus, the tape could be restored back to the root directory or be restored into a directory off the root (a root within a root).

The speed with which the files were copied to tape was truly impressive. It only took about five minutes for the Cromix partition — nearly a megabyte per minute. And everything compared OK (another sigh of relief).

The UNIX root partition was copied to tape using the rcopy utility:

```
# rcopy -t /dev/std2 /dev/ftcd          (copy)
# rcopy -tc /dev/std2 /dev/ftcd         (compare)
```

The first command performed the transfer to tape; the second performed the comparison between tape and disk.

I now had copies of the Cromix and UNIX roots on tape. There was only one thing else I needed: a version of Cromix supporting the CTD which could be booted from a floppy. The bootable Cromix floppy supplied with the system didn't have the CTD capabilities. Since I was going be to erasing the hard disk, I couldn't boot from STD0 (the Cromix root) — otherwise I would be left stranded operating system-less after the initialization took place.

So I copied the bootable Cromix floppy to a directory off the STD0 root. Then it was back to sysdef and crogen to generate a CTD capable version of Cromix (*sans* HD and SMD drivers that would take up too much space on the floppy). This version of cromix.sys was then copied into the boot disk directory. I then initialized a new floppy disk, did a makfs on it and mounted it, and copied the boot floppy directory on the hard disk to the floppy using cptree. The process was completed by then doing a wboot sfda to write a boot track to Cromix floppy. And then, just to be safe, I did the same with another floppy.

Reinitializing the Hard Disk: inithard

Now I was ready to go. I had my backup tapes, my bootable Cromix floppy with CTD drivers, and my desired disk partition table. After running shutdown (STD0 was the root), and resetting the system, I booted from the Cromix floppy (by interrupting the autoboot with CONTROL-C and issuing the RDOS command: ba).

After logging on, the first thing I did was to kill the flush utility (which is run as part of the start-up sequence). Another moment of truth was at hand. I called up inithard and when prompted for the drive to be initialized, responded with std31 (the entire hard disk). I then responded with the default responses for the next series of prompts (number of heads, number of cylinders, starting and ending cylinders to be initialized, etc.). And away it went. The surface and cylinder numbers being initialized appeared on the screen in a blurr. It was over in a matter of moments. The disk was scrubbed clean and I was truly at the mercy of the tape archives.

With that part completed, the whole point of the exercise was at hand. I was being queried by inithard as to whether or not I wanted to retain the existing disk partitions. I responded no. It then repeatedly asked for the starting cylinder of the next partition, to which I responded with the series: 85, 145, 433, 608, 679. With that it displayed the new partition table which reflected the values I had just entered. I left the alternate track table unchanged at the same location on the disk.

Installing the Cromix Partitions

With the hard disk freshly inititialized, the Cromix Root tape in the CTD, and with the floppy (sfda) as the root device, I executed the following sequence of commands:

```
# makfs std0
# mount std0 /drive
# d /drive
# ftar -vx /dev/ftcd
# ftar -vy /dev/ftcd
# wboot std31
# sync
```

I then ran shutdown and reset the computer and booted from the hard disk. It worked! The free utility informed me that there were 3,970,048 bytes (3877K) of disk space still available. Just what I had hoped for.

Setting up the Cromix data partition was a simple matter:

```
# makfs std3
# mount std3 /drive
```

Checking with free revealed that std3 had 12,011,520 bytes (11,730K) available. Again, just what I had ordered. I now had Cromix partitioned as desired. That left UNIX.

Installing the UNIX Partitions

Getting the UNIX root system from tape to disk was basically the same, except that rcopy was used in place of ftar. While running under Cromix the following commands were required:

```
# rcopy -t /dev/ftcd /dev/std2          (copy)
# rcopy -tc /dev/ftcd /dev/std2         (compare)
```

The partition to be used as a mountable drive under UNIX required a little more work. For one, the file structure on it would have to be created while operating under UNIX, using the mkfs utility (equivalent of makfs in Cromix). The *Introduction to UNIX System V for Cromemco Systems* has a chapter on reinstalling software on the hard disk. And from the example given there, and having read the *UniPlus+ System V Administrator's Manual* documentation pertaining to mkfs, I determined that command would be:

mkfs /dev/std2 (size of partition) 1 140.

The size of the partition is in terms of the number of physical sectors (1024 bytes); the 1 represents the "sector interleave factor" (taken from the Cromemco documentation); the 140 is a modulus factor (cylinder size) which varies with the drive manufacturer (supplied in the Cromemco documentation). Well, no one has ever said that UNIX is user-friendly.

At any rate, I quickly determined the partition size from Cromix by using mode std4, which reported the size as 4,970K, the value required by mkfs.

Equipped with this knowledge, I booted UNIX from Cromix-Plus by logging on as "unix". While under UNIX I made a file structure on std4 with the command:

```
# mkfs /dev/std4 4970 1 140
```

And that was that. The drive had been successfully repartitioned, and both Cromix and UNIX installed. Now I could begin with the real fun — installing the various applications packages and putting the system to work.

Other Experiences of Interest

I alluded to the fact that I had experienced some pain and

frustration along the way. The first instance arose because both of the printer ribbons supplied with the printer were dried out (despite their plastic packaging). But before I came to that unexpected realization, I had tried numerous adjustments with the print head alignment and so on, to no avail. Once I had a fresh ribbon installed the printer worked beautifully.

Another more devastating event awaited me when I first tried using the second terminal. As the electrical outlets on my power strip had been exhausted, I plugged the second terminal into a different wall outlet. In the back of my mind a little warning light when on, which unfortunately I ignored. It seems that I had heard something about "ground loops", but I figured that was a remote possibility — something that might happen to someone else. So when I turned the C-5 on and didn't hear the characteristic beep my heart skipped a beat (or two). I had one of those sinking feelings (like when you fall in a dream) when a message displayed on the screen saying "This C-5 may not be operating properly."

I inspected the system console, and to my horror, encountered a blank screen (with the cursor oddly placed in the middle-right) and an unresponsive keyboard. Turning the terminal on and off (a sure cure) had no effect. I had to reset the system without benefit of shutdown. At that point I felt that I was a certifiable MLP (Malfunction-Linked Person).

I proceeded to disconnect the offending second terminal, and said a quick prayer as I powered up the system. Thankfully, it booted. I ran check, which didn't report any errors (a tribute to flush). So far so good. I turned the printer on and tried spooling a file. Nothing. I tried redirecting output to the printer. Nothing. My worst fear had been realized — I'd blown the Octart.

In the days that followed, while the Octart was repaired, I still had use of the system (without additional terminals or printer). I had an electrician inspect the outlets, and sure enough, he located and repaired a bad ground connection. Had I gotten a bigger power strip, or had the electrical checked before installing the system, I would have saved a lot of grief. Hopefully, others can learn from my mistake.

The Beginning

Today, the system is complete and working again. It is a marvelous tool, and a joy to work with. I wish I could tell you all of the things we are doing with it, and what we plan to do. But that's another story.

I/O NEWS
PRODUCTION SCHEDULE
VOLUME V, NO. 4 TO VOLUME VI, NO. 6
(MAY/JUN '86) TO (SEP/OCT '87)

Volume/Issue	Deadline for Editorial/Ads	Deadline for Camera Ready	Issue Mailed
Vol. V, No. 4 (May/Jun '86)	May 26	Jun 2	Jun 30
No. 5 (Jul/Aug '86)	Jul 7	Jul 14	Aug 11
No. 6 (Sep/Oct '86)	Sep 8	Sep 15	Oct 6
Vol. VI, No. 1 (Nov/Dec '86)	Nov 3	Nov 10	Dec 1
No. 2 (Jan/Feb '87)	Jan 6	Jan 13	Feb 3
No. 3 (Mar/Apr '87)	Mar 10	Mar 17	Apr 7
No. 4 (May/Jun '87)	May 5	May 12	Jun 2
No. 5 (Jul/Aug '87)	Jul 7	Jul 14	Aug 4
No. 6 (Sep/Oct '87)	Sep 8	Sep 15	Oct 6

INSIDE CROMIX

INSIDE CROMIX is an open forum on both eight-bit and 16-bit versions of Cromix. The subject matter is directed towards helping Cromix users derive more from their systems. Members' contributions are invited. INSIDE CROMIX is edited by Jordan Siedband, who can be reached at 5017 Fairview Lane, Skokie, IL 60077, (312)674-1175.

A UNIVERSAL MENU PROGRAM

I have included a 'C' listing of a menu program which will call and run any set of programs listed in an ASCII file.

To load the menu file, use screen, screen -n, or ce menufile, where menufile represents the name of your menu. The file should be constructed as follows:

1st Line = Title (to be shown at top of menu)
2nd Line = Selection Description1 (e.g., Enter Payroll Hours)
3rd Line = Program Name1 (e.g., hours.bin)
4th Line = Selection Description2
5th Line = Program Name2
. . . (MAX 26 programs and descriptions)

Be sure that the last line is null, i.e., the cursor is at the beginning of the last line in the file.

To call the menu use the command: menu menufile

This should work for any related set of programs. To use this program correctly, each of your 'C' programs must have been complied as XXX.bin and should have as its last call

exec("menu.bin","menufile",nullchar);

where nullchar has been defined as in line 17:

char *nullchar[] = {'\0'};

As always, if you send an initialized disk with enough return postage and a self-addressed mailing label, I will return compiled copies with source code and help file. If you are using other than Cromemco terminals, study lines 22-28, read your terminal manual, and give me the correct values for your terminal.

PROGRAM LISTING
==========

```
 1                /*      menu: GENERAL main menu  4/06/86        */
 2                /*      Jordan Siedband (312)674-1175           */
 3                /*      5017 Fairview Lane                      */
 4                /*      Skokie,IL 60077                         */
 5
 6   #include <modeequ.h>
 7   #include <stdio.h>
 8   #define MAXNO   26
 9
10   char  versn[]="\375\355\375\355\0\0\0\0\0\0\
11   menu GENERAL main menu  J.Siedband (312)674-1175  4/06/86\r\n";
12   char    *ctl="%s";
13   char    *title;
14   char    *endmsg= "LEAVING %s";
15   char    *loadmsg="Loading %s";
16   char    buff[80],nbuff[80*MAXNO];       /* input and string buffers */
17   char    *nullchar[]={'\0'};
18
19                /* These are set up for CROMEMCO TERMINALS */
20                /*   Replace with calls for your terminal  */
21   /*********************************************************************/
22   char    clrall[6]="\033E";      /* replace with screen clear string */
23   char    clreop[6]="\033J";      /* replace with clear to eop string */
24   char    clreol[6]="\033K";      /* replace with clear to eol string */
25   char    crlead[6]="\033F";      /*   cursor leadin, 1B 45 0 for C5  */
26   char    offset=' ';             /*    cursor offset, shown for C5   */
27   char    standout[6]="\033dP";   /*          start highlight         */
28   char    standoff[6]="\033d8";   /*           end highlight          */
29   /*********************************************************************/
30
31   FILE    *fp;
32
33   struct
34   {
35           char *action;   /*  line to display     */
36           char *file;     /*  program path name   */
37   }progs[MAXNO];
38
39   main(argc,argv)
40   int     argc;
41   char    *argv[];
42   {
43           int c,i,j,k,nmbr,pos;
44           char date[4],dat[9];
45           long tt;
46
47                   /* first read title from menu ascii file        */
48           if((fp=fopen(argv[1],"r"))==NULL)
49                   syserr("\007NO SUCH MENUFILE\n");
50           clrbuff();      /* null out input buffer        */
51           fgets(buff,80,fp);
52           nmbr=j=0;
53           i=-1;
54           title=&nbuff[0];        /* address of title=start of nbuff */
55           while(buff[++i]!='\0' && buff[i]!='\n' && buff[i]!='\r')
56               nbuff[j++]=buff[i];
57           nbuff[j++]='\0';
58                   /* now get actions and filenames  */
59           clrbuff();
60           while(fgets(buff,80,fp)!=NULL)
61           {
62                   /* address of action=current address of nbuff */
63               progs[nmbr].action=&nbuff[j];
64               i=-1;
65               while(buff[++i]!='\0' && buff[i]!='\n' && buff[i]!='\r')
66                   nbuff[j++]=buff[i];
67               nbuff[j++]='\0';
68               clrbuff();
69               fgets(buff,80,fp);
70                   /* address of filename=current address of nbuff */
71               progs[nmbr++].file=&nbuff[j];
72               i=-1;
73               while(buff[++i]!='\0' && buff[i]!='\n' && buff[i]!='\r')
74                   nbuff[j++]=buff[i];
75               nbuff[j++]='\0';
76               clrbuff();
77           }
78           fclose(fp);
79
80           pos =(80 - strlen(title))/2;    /* centers header       */
81           cursor(1,1);
82           printf(ctl,clrall);
83           getdate(date);
84           i=date[0];                      /*   get date from system  */
85           j=date[1];                      /*   in the form  4/15/83  */
86           k=date[2];
87           tt=1000*(1000*i+j)+k;
88           sprintf(dat,"%2d/%02d/%02d",i,j,k);
89           dat[8]='\0';
90
91           cursor(pos,1);
92           printf("%s",title);
```

```
93          cursor(36,2);
94          printf("%s\n",dat);             /* print page heading  */
95          cursor(36,3);
96          printf("========\n");
97
98          if (nmbr <15)   /* if less than 15, prints in a single col */
99          {
100             for (i=0;i<nmbr;i++)
101             {
102                 cursor(20,i+5);
103                 printf("%c:  %s\n",i+65,progs[i].action);
104             }
105             cursor(33,nmbr+6);
106             printf("0 = out\n");
107         }
108         else            /* otherwise prints in two columns      */
109         {               /*      do not exceed 26 items          */
110             k=nmbr/2;
111             for (i=0;i<k;i++)
112             {
113                 cursor( 1,i+6);
114                 printf("%c: %s\n",2*i+65,progs[2*i].action);
115                 cursor(41,i+6);
116                 printf("%c: %s\n",2*i+66,progs[2*i+1].action);
117             }
118             if (nmbr % 2) /* if odd number, print out extra entry */
119             {
120                 cursor(20,k+6);
121                 printf("%c: %s\n",nmbr+64,progs[nmbr-1].action);
122                 k++;
123             }
124             cursor(36,k+7);
125             printf("0 = out\n");
126         }
127
128 ask:    if (nmbr < 15)  cursor(33,nmbr+8);
129         else            cursor(33,k+9);
130         printf(ctl,clreol);
131         printf(ctl,standout);
132         printf(" Your Selection? ");  /* allows for 26 selections */
133         printf(ctl,standoff);
134         putchar(' ');
135
136         setmode(STDOUT,MD_MODE1,0,ECHO);        /*  no echo  */
137         setmode(STDOUT,MD_MODE1,RAW,RAW);       /* immediate */
138         while((c=getchar())==EOF);
139         setmode(STDOUT,MD_MODE1,ECHO,ECHO);
140         setmode(STDOUT,MD_MODE1,0,RAW);
141             c=toupper(c);
142         if (c == '0')
143         {
144                 cursor(1,4);
145                 printf(ctl,clreop);
146                 sprintf(buff,endmsg,title);
147                 cursor(40-strlen(buff)/2,10);
148                 printf("%s\n",buff);
149                 exit(0);
150         }
151         i=c-'A';
152         if (i<0 || i>nmbr-1) goto ask;
153         cursor(1,4);
154         printf(ctl,clreop);
155         sprintf(buff,loadmsg,progs[i].file);
156         cursor(40-strlen(buff)/2,10);
157         printf("%s\n",buff);
158         /*    load and run  progs[i].file       */
159
160         exec(progs[i].file,nullchar);
161 }
162 strlen(a)  /* returns length of string a not including final '\0'  */
163 char  *a;
164 {
165         int i;
166
167         for(i=0;*(a+i) != '\0';++i);
168         return(i);
169 }
170 clrbuff()
171 {
172     int i;
173
174     for(i=0;i<80;i++)   buff[i]='\0';
175 }
176 cursor(x,y)     /*   places cursor at col,row    home is 1,1    */
177 int x, y;
178 {
179         printf(ctl,crlead);
180         putchar( y + offset -1);
181         putchar( x + offset -1);
182 }
```

```
      utility:      MENU

      purpose:      Run Menus of Programs

  user access:      all users permitted access to menufile

      summary:      menu menufile

    arguments:      menufile, name of menufile

      options:      none

     location:      /bin directory
```

Description

A typical menufile is constructed using screen, screen -n, or ce as follows:
ce menufile

```
line 1: TITLE           (printed as header to the menu)
line 2: DESCRIPTION 1   (To show action to be taken)
line 3: PROGRAM NAME 1  (pathname of executable bin program)
line 4: DESCRIPTION 2
line 5: PROGRAM NAME 2
        etc., notice there are always an odd number of lines, maximum
              number of descriptions and programs=26.
```

For example, a payroll menu might be the file "paymenu":

```
PAYROLL PROGRAMS
NEW EMPLOYEE
newemp.bin
MODIFY EMPLOYEE
modify.bin
ADD HOURS
hours.bin
CALCULATE TAXES
calculat.bin
WRITE CHECKS
checks.bin
CHECK REGISTER
register.bin
QUARTERLY REPORT/W-2s
qrtlyw2.bin
CLEAR EMPLOYEE QRTR/YEAR
clearemp.bin
```

These eight entries might be protected by putting the file "paymenu" and the *.bin programs in the payroll directory with owner access only. To call the menu properly, the proper owner should be in the payroll directory and call

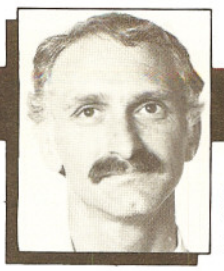
```
menu paymenu
```

Each of the bin programs should have, early in the source code:

```
char *nullchar[]={'\0'};
```

and upon exit:

```
exec("menu","paymenu",nullchar);
```



C-10 ENCOUNTERS

C-10 ENCOUNTERS is a regular column directed to users of Cromemco's personal computer, the C-10. It is edited by Dr. Tom Beer, of Applied Environmetrics, located at 118 Gordon St., Balwyn, Victoria 3103, Australia. Dr. Beer can be reached by phone during business hours at (03) 817-2571. Submit editorial directly to Dr. Beer.

MAILBOX

Regular readers of this column will be aware that I have a bit of an aversion to hardware and consider the act of screwing in a CFD disk drive to be the height of electronic achievment. Because of this I am always extremely impressed by correspondents who regularly seem to rip out the insides of their C-10 and replace esoteric chips. Robert Lagace of the Universite Laval, Ste-Foy, Quebec seems to fall in this category and he wrote me an interesting letter pointing out that "It is possible to add a second 5.25 inch floppy disk drive to the C-10 disk drive cable. The only thing you need is a second floppy drive, a two drive cabinet without power supply and about $10 of hardware. I do not know why Cromemco does not offer two drives in one cabinet. The cable is almost as expensive as the drive. Drives other than the TANDON TM100-2 may be used. We also successfully made a cable to connect an 8" floppy disk drive. The 8" drives are interesting but do not show any speed gain as would have been expected. The CDOS disk driver must be improved."

Other interesting hardware projects that he describes include:

- Implementation of a french character generator by plugging a small PC board into the socket of the character generator. The PC board has both the original and the french chips on it and the character set needed is selected by a switch put in the back of the C-10.
- Connection of the C-10 to data acquisition systems via the modem port.

On the software side of things he strongly recommends Turbo Pascal (US$70 from Borland International) and points out that it works fine with CDOS 3.07 (Release 5 and 6) but requires a patch to work with previous versions. He has written programs in Turbo Pascal to autodial remote data loggers, interrogate them, retrieve the data and store them onto the C-10 disk. Turbo Pascal permits one to program in Pascal special functions that must be programmed in assembly language with other languages and thus speeds up program development by a factor of three.

On printers and Writemaster he writes: "There exist other ways to solve the problem of the reset code sent to the printer driver by Writemaster. The one I used disables that reset by patching Writemaster byte 7F at location 569C by byte 00. It took me 3 hours to find the location of this call. If you look in a C-10 User manual dated February 1984 or later then you will find that 7F resets the printer. Because the 12 pitch character command is a two character command 'ESC,K' (1B,4B), it cannot be implemented simply in Writemaster. By disabling this reset you may write any program that selects your choice. You need to run it when you start your C-10 and your selection is good for all your sessions.

The problem mentioned by Saul Weitz (one line too many on each page) seems to exist only with the 'generic' printer driver. I did not encounter it when using the 'Epson FX80' and the 'Brother HR15' drivers.

Writemaster still has bugs. ALIGN produces garbage when aligning through text containing bold or underline. The left margin setting requires a number greater than or equal to 1. A zero setting is equivalent to a setting of 255."

Bill Jaenicke drew my attention to the fact that ProtoMatrix Software Development has produced ProPrint to provide general printer support for the C-10 and which, hopefully, will overcome the ubiquitous problems of interfacing printers and Writemaster. I use a CLQ (actually a Smith-Corona TP-I) so have had no opportunity to use ProPrint. Readers comments about it, for good or ill, are invited.

Saul Weitz of New York wrote again to comment on some problems with Procall and Prolink. He warns that the file Pcxfer.com on Prolink is not the same as the identically named file on the Procall disk. He also writes: "...although I have a disk clearly marked C-10 version, and the documentation is marked likewise, the default terminal for the program is the Cromemco 3102. Before the program works properly, you must go into PcMod on the Procall disk to change the terminal." This seems curious. The C-10 was built to emulate the Cromemco 3102 so that there should be no need to change things.

Terminals

What does emulate the 3102 mean? It means that the escape sequences that control the C-10 should be the same as the escape sequences that govern the 3102, and this is indeed the case. The most important of these have been covered in earlier columns; Esc"E" clears the screen, Esc"H" moves the cursor to the top left of the screen. A complete list is given in Appendix E of the C-10 Technical manual.

In the case of the 3102 I belive that there is a little chip inside it and when an escape sequence comes from the computer to the terminal it acts on it. In the C-10, screen handling is done by the operating system and the screen emulation behaviour is specified in the C-10 software. This difference led to an interesting difference in the little demonstration train that was supplied on C-10 systems disks prior to Release 4. On the large Cromemco systems the train demonstration was a data file and one could get a picture of a train on a 3102 terminal with TYPE TRAIN.DAT as the command. The C-10 operating system produces a visible representation of Esc sequences when they are typed, Esc"E" appearing as v E for example, so that the train program on the C-10 had to be a program TRAIN.COM, with commands to print things like Esc"E" issued from the program.

Many C-10 users have had prior experience with 3102 terminals and there is a tendency to believe that the standard way to toggle the cursor from an Sbasic program is with a command along the lines of @Chr$(27);"Z". During a recent meeting with Dr. John Zeigler he pointed out that according to the Sbasic manual the proper way to do this is by using print parameters. He also pointed out that @\0,38,0\ is supposed to turn the cursor on whereas @\0,39,0\ is supposed to turn it off. In fact, they both do the same thing and toggle the cursor and are thus both equivalent to the Esc"Z" sequence. The fault in this case lies not with Sbasic but with the operating system. The Basic interpreter transforms this form of the print command into CDOS system call 142 (8Eh) with the parameter p1 placed into the D register. And the error lies in the operation of this system call. To guarantee a cursor on, use an Esc"r" sequence and Esc"q" for cursor off (lower case r and q).

Ms. K. Baulch ask whether there is such a thing as a C-10 reset. She says " I do get into the most difficult situations sometimes where things hang up, the screen goes blank and/or

whatever I press has no effect on anything but the beep. This can happen anytime doing anything. The Technical Manual says that Control-Shift R for Reset operates locally at all times and so does Esc"V" but I haven't had any success with this." Neither have I. The only way to get a good reset is to turn the C-10 off at the power switch, wait for five seconds and then turn it on again.

Peter Rogers of Deakin University wondered about the operation of the Esc"G" sequence which is supposed to read the character at the cursor location. It appears to be unusable in Basic but if it is tested with an Assembly language program then it appears to act in much the same way as pressing a keyboard character whilst a program is running on the C-10. The last keypress is stored and returned on the screen next to the A. prompt when the program finishes.

John Zeigler's 14 year old son David is also a keen C-10 user and he sent me a disk with a couple of programs on it. One of them requires you to steer a spaceship through a tunnel and I have reproduced it below. It has some nice examples of the use of timed inputs, cursor positioning, and graphics.

```
 200   Set 14,1
 300   Dim Zz(100),A$(0)
 400   Set 0,-1
 500   Randomize
 600   B=0
 700   Close
 800   Open\1\"$C0"
 900   @\1,0\
1000   Set 5,0
1100   Input"Enter speed of descent (1 to 10): ",S
1200   If S<1 Or S>10 Then @"Try again" : Goto 1100
1300   S=S/10
1400   X=Int(54*Rnd(1))
1500   @\1,38\
1600   @\1,0\Chr$(27);"dP";
1700     For I=1 To 20
1800     @\1,X,I\Chr$(27);"d@       ";Chr$(27);"dP";
1900     X=X+B
2000     Zz(I*2)=X-1
2100     B=Int(2*Rnd(1))+1
2200     C=Int(2*Rnd(1))+1
2300     If C=2 Then B=-B
2400     If X+B<=0 Then Goto 2100
2500     If X+B>=54 Then Goto 2100
2600     Next I
2700   @\1,38\
2800   Rem                        THE SHIP!!!!!!!
2900   @\1,38\
3000   Y=Zz(2)+2
3100   @\1,Zz(2)+2,1\"¦^¦";
3200     For I=1 To 20
3300     On Error Goto 3600
3400     Set 5,S*5
3500     Get\1\A$
3600     On Error Stop
3700     @\1,Y,I\"    ";
3800     If A$="," Then Y=Y-1
3900     If A$="." Then Y=Y+1
4000     @\1,Y,I+1\"¦^¦";
4010     If I=20 Then 5500
4100     If Y<=Zz(I*2) Then Goto 4500
4200     If Y+1>=Zz(I*2)+6 Then Goto 4500
4300     Next I
4400   Goto 5500
4500   @ Chr$(7);
4600   @\1,38\
4700   Goto 6000
4800   @\1,0\
4900     For I=1 To 20
5000     @\1,I,I\Chr$(27);"d@";
5100     @\1,I+6,I\Chr$(27);"dP";
5200     Next I
5300   End
5400   List"tunnel"
5500   @\1,0\
5600   @\1,38\
5700   @"      YOU DID IT!!!!"
5800     For I=1 To 100
5900     Next I
6000   Set 5,0
6100   Input"Do you want to play again?",K$
6200   If K$="y" Or K$="Y" Then Goto 200
```

SUPER CHALLENGING TEST

All the above letters and discussions have raised lots of questions in my mind, many of which I am unable to answer. Here then is my 1986 super challenging test:

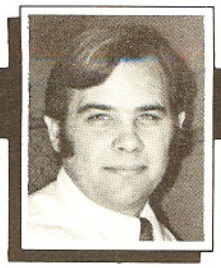
1. Describe any way of returning from CDOS to CROS without turning off the machine.

2. Show how to get a Basic program to read the character at the cursor location, using Esc"G", and return the value in a variable. Show how to get an assembly language program to use the result from this escape sequence.

3. Explain what Set 14,1 does in David Zeigler's program, and why the Sbasic manual says that system parameter 14 cannot be set.

4. In *I/O NEWS Volume IV Number 6* Mike Bennett's Basic program to poll the I/O port was given. It had to diable interrupts to work. Describe the operation of the interrupt service routine when interrupts are not disabled.

My own score on this test is 0.5 out of 4. I did find a way toget the CROS ';' prompt up from CDOS C2.65 (Release 4) without turning off the machine but it required both a program to jump to the CROS service routine, try to establish external connection, fail and wait for a keypress from the operator. I gave myself half a mark and decided that there has to be a better way. I have a feeling that if I could answer question 4 then question 2 would be easy. And finally, I could find no difference in the operation of the program with Set 14,0. maybe it really does not do anything? Please write if you can answer any of these questions.



TEC TIPS

TEC TIPS is a regular column aimed at providing hints for keeping systems up and running. It will not attempt to deal with specific engineering applications or non-standard configurations. TEC TIPS is edited by Richard Quinn, owner of QUINN TEAM, a Southern California computer service firm. Telephone (818) 889-4819

Ftar Check Sum Errors

In the last *Tec Tips* I reported a problem with ftar check sum error messages that I thought were coming from the flush command.

Unfortunately I don't remember the exact hardware/software configuration of the system giving trouble and have been unable to help Cromemco reproduce the symptoms on their machines. It could be that it was a function of a defective memory card or some other system specific problem and not a utility problem. At the time it was so repeatable that I didn't think there would be any problems reproducing it. So this may not be a problem at all. If any of you have had similar problems please let me know as I would like to know where this phantom went!

Floppy Disk Drivers Under Cromix-Plus

The current release of Cromix-Plus, 30.05, has a problem with the floppy drivers when a seek error is encountered. The driver does not recover on any of the following retries (the status register appears to always remain the same). I have tested this on the bench by causing a seek error and then disabling the read circuitry of the FDC card and the error status does not change to read errors during retries. Therefore, I know that if the controller is not reading the disk it cannot possibly recover from a seek error (without reading how would the system know what track it is on?). The problem was first noticed with TM-848 drives that had greasy rails, causing the original seek error and not recovering on following retries.

I set the drive's retries to 1 (normally 10) using mode (you can also reset hard and soft errors to 0 using mode). I do this so that when an error does occur it does not seem to take forever for the system to come back and give up on the job.

I prevent seek errors in the first place by being certain that the drives are clean, VERY clean, especially the rails on 8'' Tandon floppy drives. That helps keep seek errors from occurring in the first place. Cromemco is aware of this problem as has fixed it in the upcoming release of Cromix-Plus.

Octarts and IOP/Quadarts

There is an easy way to test Octarts and IOP/Quadart setups to give a simple determination wether or not the hardware is operating, or at least alive. While this method is not completely fool-proof it will give you an indication of the cards function.

From now on I will refer to the Octart and only mention the IOP/Quadart set up if it is different. So consider these instructions the same for both card configurations.

The Octart is a micro computer in its own right. If fact it is a complete Z80 system whose only function in life is to handle the terminal I/O. The operating instructions are loaded from the host computer during the boot of the system using the /etc/iostartup.cmd file as the driver to load appropriate software. This software then becomes the cards basic set of instructions.

Prior to loading this instruction program the card comes up using the IOP Monitor ROM on the card. The function of this Monitor is to cause the Z80 to come to life in a known state on power up and set up the card to receive the iostartup program. But prior to the iostartup loading, the monitor ROM allows the card to be reset and controlled from the system and also monitors the serial ports with an automatic baud program in the same manner as RDOS.

It is this auto baud program that can come in handy for quick hardware checks. Any terminal running at 9600 baud or less will produce an ''IOP Monitor'' message if the RETURN key is pressed several times allowing the baud rate to be selected. This allows certain RDOS-like commands like DM begin-end where begin is a beginning address and end is the ending address to display. Note that I said 9600 baud as I have only been able to get the monitor at 9600 and lower baud rates. 19200 does not seem to work.

When a particular port (on an Octart you have eight (8) and with an IOP/Quadart there can be up to twelve (12) with three Quadarts) is selected, all other ports on that card are disabled. But you can simply hit reset on the computer and connect to another port to test it. So a simple hardware test would be to check that all terminals were running at 9600 baud, and then, one at a time, hit reset and press return several times on each

of the connected terminals to see that the IOP Monitor message appears. If it appears on all terminals before, but not after, the system boots, the problem is likely in the iostartup.cmd file, ttys file, a dev file, or the way Cromix is "GENed."

Alternately, all ports can be tested by moving a single terminal to each port, one at a time, hitting reset in between moves and checking for the IOP monitor program on each port. In the case of an Octart this is each connector on the splitter cable; on a Quadart this is J2, J4, J6, and J8.

Note that if your system is set to automatically boot on reset, as in the case of an all IOP or Octart system, you do not want to be constantly hitting reset in the middle of a boot sequence for fear of your hard disk's data. Pulling an essential card off the bus, like the DPU or FDC, will stop this auto booting without affecting the IOP Monitor. In fact, the Octart will respond even if it is the ONLY card on the bus. This is because it is a fully self-contained microcomputer and needs only power for the monitor to operate.

I use this technique often to make a quick determination of the hardware connections, especially where one terminal does not work.

STDC Hard Disk Parameters.

The newest version of the STDC initialization program, inithard version 30.07, allows you to initialize almost any type of ST-506 hard disk drive. I have done many different types and will pass on the parameters I use.

The key bits of information needed are as follows: 1) total number of cylinders; 2) total number of surfaces; 3) number of alternate tracks; 4) starting track of precomp. Partition information is also needed but is largely up to the user. If inithard cannot determine what type hard disk drive you have (done by reading the label which does not exist yet on a never been initialized hard disk) it makes a guess so that the default values may be way off from that needed.

The following is a list that I would very much like to have grow with time. That is, if any of you have drives that can be added to this list please send me the information and I'll constantly update this list for all known drives. I will not include number of alternate tracks as I find it best to supply that number manually. I give about 6 more tracks than known bad tracks. These 6 extras will provide for future use if needed. Default number of alternate tracks is often far more than needed and wastes usable data area.

Manufacturer	Model #	Cap.	# of Cyl.	# of Heads	Precomp
Control Data	94155-57	50Mb	925	6	0
Hitachi	DK511-5	50Mb	714	7	256
IMI	5018	20Mb	306	6	214
Maxtor	XT-1065	67Mb	918	7	918
Maxtor	XT-1085	85Mb	1024	8	1024
Maxtor	XT-1105	83Mb	918	11	918
Maxtor	XT-1140	143Mb	918	15	918
Maxtor	XT-2085	85Mb	1224	7	1224
Micropolis	M1304	50Mb	830	6	829
Rodime	50240	25Mb	640	6	640
Shugart	712	10Mb	306	4	200
Tandon	TM-501	6Mb	306	2	306
Tandon	TM-502	12Mb	306	4	306
Tandon	TM-503	19Mb	306	6	306
Tandon	TM-251	6Mb	306	2	306
Tandon	TM-252	13Mb	306	4	306
Vertex	V-150	50Mb	987	5	986

The partitions can be defined by the user. As a rule of thumb I define the partitions in 20 to 30 megabyte sections so that they will fit on a CTD tape. But you may opt for smaller partitions so you can use a product like Fastback or ftar to back up a segment.

To calculate the size of a partition in megabytes you can figure about how many divisions in the drive to make each one the desired size, and then divide the number of cylinders by that number to get the number of tracks. For instance, a Maxtor XT1185 is about 85 megabytes. If I wanted each partition to be about 20 megabytes then that would divide the 1024 cylinders into about fourths. 1024 cylinders divided by 4 would equal 256 cylinders per partition. The first partition would be at cylinder 256, the second at 512, the third at 768. This would produce a drive with STD0 on cylinders 0-255, STD1 on cylinders 256-511, STD2 on cylinders 512-767 and STD3 on cylinders 768-1024. Each partition would contain about 21.25 megabytes.

The choice for partitions is yours and I would recommend using more partitions, not less, as this makes clean up of a bad disk partition fast and easy. Partitions can always be mounted under any name and thus will fit into the directory tree where desired.

Diskinfo When Everything is OK

A new utility called diskinfo is included on all systems now. It should be your first order of business upon receiving a new hard disk. Don't wait to run this utility when the system is trashed as it will likely not work. Run the utility and direct its output to a file called /etc/sys.doc to keep a record of the disk alternate tracks and partitions. This will be of great value if the hard disk is ever initialized. Print the file and keep a copy with your system books.

The command diskinfo std31 > /etc/sys.doc will provide a file that can be screened and printed for the first hard disk on the system. The same command line for std63 will cover the second hard disk. Even if you don't understand all of the information from diskinfo, a qualified service person will find it very helpful, especially if you need to retrieve data.

I also put information about the printers in use, terminals and baud rates and any other system notes in this sys.doc file. I always make certain there is a current hard copy in my system papers, usually the system log book. These notes are invaluable when restarting a crashed system or in simple trouble shooting of problems.

32K CLASSROOM

32K CLASSROOM is a regular column aimed at explaining programming techniques using Cromemco Structured BASIC. It is edited by Bernie Thomas, President of Jakes Manufacturing Corp., P.O. Box 23050, Nashville, TN 37202. Submit any editorial contributions to I/O NEWS in care of the 32K CLASSROOM.

Anyone who has had to figure the FICA and withholding taxes for a number of hourly employees, whose gross pay varies from week to week, certainly must appreciate the marvels of a computer. If you have written, or thought about writing a payroll program, you have no doubt encountered the problem of taxes.

There are two different ways to figure withholding taxes. You can use the "alternative formula tables for percentage method withholding," which can be found in the Internal Revenue Services Publication 493. I chose to use Table 1 in Circular E, which is for a weekly payroll period.

Since this is a discussion of taxes and not payroll per se, I have written a short input program to demonstrate the method of calculating the correct taxes. In actual use, the tax table routine would be included as part of a larger payroll program.

```
10   Dim F$(7) : F$="#,###.##"
20   On Error Stop : Input"Enter the WEEKLY GROSS PAY )",Gross'pay$
30   If Gross'pay$="end" Then Stop
40   On Error Goto 20 : Gross'pay=Valc(Gross'pay$)
50   Input"ENTER '1' if single - '2' if married )",Marital'status$
60   If Marital'status$#"1" And Marital'status$#"2" Then 50
70   On Error Stop : Input"Enter NUMBER of DEPENDENTS )",Dependents$
80   On Error Goto 70 : Dependents=Valc(Dependents$)
90   T'gross=Gross'pay-(Dependents*20.77)
100  On Val(Marital'status$) Gosub Single'tab,Married'tab
110  Fica=Gross'pay*0.0715
120  Net'pay=Gross'pay-Fica-Wh
130  @ : @ Tab(10);"GROSS PAY";Tab(25); : @ Using F$,Gross'pay
140  @ Tab(10);"FICA";Tab(25); : @ Using F$,Fica
150  @ Tab(10);"WITHHOLDING";Tab(25); : @ Using F$,Wh
160  @ Tab(10);"NET PAY";Tab(25); : @ Using F$,Net'pay
170  @ : @"ENTER 'end' TO STOP"
180  Goto 20
190  Stop
200 *Single'tab
210  If T'gross)28 And T'gross(=87 Then Wh=(T'gross-28)*0.12
220  If T'gross)87 And T'gross(=192 Then Wh=7.08+((T'gross-87)*0.15)
230  If T'gross)192 And T'gross(=302 Then Wh=22.83+((T'gross-192)*0.19)
240  If T'gross)302 And T'gross(=457 Then Wh=43.73+((T'gross-302)*0.25)
250  If T'gross)457 And T'gross(=577 Then Wh=82.48+((T'gross-457)*0.3)
260  If T'gross)577 And T'gross(=687 Then Wh=118.48+((T'gross-577)*0.34)
270  If T'gross)687 Then Wh=155.88+((T'gross-687)*0.37)
280  Return
290 *Married'tab
300  If T'gross)50 And T'gross(=199 Then Wh=(T'gross-50)*0.12
310  If T'gross)199 And T'gross(=398 Then Wh=17.88+((T'gross-199)*0.17)
320  If T'gross)398 And T'gross(=490 Then Wh=51.71+((T'gross-398)*0.22)
330  If T'gross)490 And T'gross(=600 Then Wh=71.95+((T'gross-490)*0.25)
340  If T'gross)600 And T'gross(=710 Then Wh=99.45+((T'gross-600)*0.28)
350  If T'gross)710 And T'gross(=930 Then Wh=130.25+((T'gross-710)*0.33)
360  If T'gross)930 Then Wh=202.85+((T'gross-930)*0.37)
370  Return
```

Line 90 of the program calculates the TAXABLE GROSS PAY, which I have called T'gross. The allowance of 20.77 times each dependent changes, as do the figures in the table and the FICA percentage. It is therefore necessary to edit the routine as taxes change.

If you are going to write a payroll program for someone who would not have the ability to edit the changes, I suggest you substitute variables for the actual figures, and write the routine to read the current values from a file which the user can update when needed. In order to simplify the update, I wrote a program which replicates the table as it appears in the IRS Circular. This makes it much easier for the user to check the figures. Using variables has an additional advantage. It allows you to figure both single and married with the same code. Actually you could figure all eight tables, biweekly, semiweekly, etc., from the same code, since the only thing that changes is the figures.

```
100  Dim S$(159)
110    For Record=0 To 1
120    Gosub Crt'clear
130    L=1 : C=3 : Gosub Loc : @"TABLE 1 - ";
140    @"If the Payroll Period With Respect to an Employee is Weekly"
150    If Record=0 Then  Do : Who$="SINGLE"
160      Else : Who$="MARRIED" : Enddo
170    @ : @"(a) ";Who$;" person - including head of household:"
180    @"If the amount";Tab(20);"The amount of income tax"
190    @"of wages is:";Tab(20);"to be withheld shall be:"
200    @ : @ : @"Over-      But not over-";Tab(42);"of excess over-"
210  *Line1 : By=0 :.  For Item=1 To 3
220      L=9 : C=12*(Item-1)+1 : Gosub Loc
230      If Item=2 Then @"-";
240      If Item#3 Then Input"$",No$
250      If Item=3 Then  Do : Input"",No$
260        L=9 : C=C-31+Len(No$) : Gosub Loc : @"X" : Enddo
270      If Item=2 Then  Do : L=10 : C=1 : Gosub Loc : @"$";No$
280        L=10 : C=45 : Gosub Loc : @"-$";No$ : Enddo
290      If Item=1 Then L=9 : C=45 : Gosub Loc : @"-$";No$
300      S$(By,By+7)=No$(-1) : By=By+8 : Next Item
310  *Line2'5
320      For Line=2 To 6
330        For Item=1 To 3
340        If Item=3 Then L=L-31 :C=C-31+Len(No$)+2 : Gosub Loc : Goto 360
350        L=8+Line : C=(12*Item)+1 : Gosub Loc
360        If Item=1 Then @"-";
370        If Item#3 Then Input"$",No$
380        If Item=3 Then  Do : Input" plus ",No$
390          L=8+Line : C=C-25+Len(No$) : Gosub Loc : @"X" : Enddo
400        If Item=1 Then  Do : L=8+Line+1 : C=1 : Gosub Loc : @"$";No$
410          L=8+Line+1 : C=45 : Gosub Loc : @"-$";No$ : Enddo
420        S$(By,By+7)=No$(-1) : By=By+8 : Next Item
430      Next Line
440  *Line7
450      For Item=2 To 3
460      If Item=3 Then L=L-31 : C=C-31+Len(No$)+2 : Gosub Loc : Goto 490
470      L=15 : C=(12*Item)+1 : Gosub Loc
480      If Item=1 Then @"-";
490      If Item#3 Then Input"$",No$
500      If Item=3 Then  Do : Input" plus ",No$
510        L=15 : C=C-25+Len(No$) : Gosub Loc : @"X" : Enddo
520      S$(By,By+7)=No$(-1) : By=By+8 : Next Item
530    L=20 : C=5 : Gosub Loc
540    @"IS YOUR INFORMATION CORRECT? (ans y/n) )"; : Get\0\I$(0,0)
550    If I$="n" Or I$="N" Then Goto 120
560    If I$#"y" And I$#"Y" Then 530
570    Open\1,160\"taxtable.dat"
580    Put\1,Record\S$(-1) : Close
590    Next Record
600   L=23 : C=1 : Gosub Loc : Stop
1000 *Crt'clear : @ Chr$(27);"+"; : Return
1010 *Loc : L=L+31 : C=C+31 : @ Tab(0);"";
1020   @ Chr$(27);"=";Chr$(L);Chr$(C); : Return
5000 *Sav : Save"taxtable1.pyr"
```

What this program does is prompt the user for an input at a location on the screen, which as I have stated, replicates the tables in Circular E of the Employer's Tax Guide. First of all, let me note that the cursor locating code which appears on line 1020 is for the Televideo Model 950 and must be changed to suit your particular terminal.

As you can see in Line 110, the program is a "for-next loop" which will run twice: the first time for a "SINGLE Person" and a second time for a "MARRIED Person." This is evident in lines 150 and 160. Each time the program is run it will build a 160-byte string, which I have called S$. This is done in lines 300, 420, and 520. After the string is completed and okayed, it is stored in a file which I have called taxtable.dat. The first string, which is for a single person, is stored in record 0, and the second, which is for a married person, is stored in record 1. This is done in Line 570.

Usually, in an input program such as this, I include a review and correction routine which would allow the user to correct any portion of the entry. Such a routine is fairly long, and since this program is normally used only once each year, I chose to let the user re-enter the entire string in the event of an error. Thus Line 550 sends the user back through the entire entry pro-

cess if the answer to Line 540 is "no."

The following program is the same as the first program except I have substituted the single routine "Tab" for both "Single'tab" and "Married'tab." Line 190 opens the file taxtable.dat and reads the appropriate record to the variable S$. Line 200 uses a loop to assign the value to 20 arithmetic variables called S(1) through S(20).

```
 10   Dim S$(159),F$(7) : F$="#,###.##"
 20   On Error Stop : Input"Enter the WEEKLY GROSS PAY )",Gross'pay$
 30   If Gross'pay$="end" Then Stop
 40   On Error Goto 20 : Gross'pay=Valc(Gross'pay$)
 50   Input"ENTER '0' if single - '1' if married )",Marital'status$
 60   If Marital'status$#"0" And Marital'status$#"1" Then 50
 70   On Error Stop : Input"Enter NUMBER of DEPENDENTS )",Dependents$
 80   On Error Goto 70 : Dependents=Valc(Dependents$) : On Error Stop
 90   T'gross=Gross'pay-(Dependents*20.77)
100   Record=Val(Marital'status$) : Gosub Tab
110   Fica=Gross'pay*0.0715
120   Net'pay=Gross'pay-Fica-Wh
130   @ : @ Tab(10);"GROSS PAY";Tab(25); : @ Using F$,Gross'pay
140   @ Tab(10);"FICA";Tab(25); : @ Using F$,Fica
150   @ Tab(10);"WITHHOLDING";Tab(25); : @ Using F$,Wh
160   @ Tab(10);"NET PAY";Tab(25); : @ Using F$,Net'pay
170   @ : @"ENTER 'end' TO STOP"
180   Goto 20
190 *Tab : Open\1,160\"taxtable.dat" : Get\1,Record\S$ : Close\1\
200   By=0 :   For X=1 To 20 : S(X)=Val(S$(By,By+7)) : By=By+8 : Next X
210   If T'gross)S(1) And T'gross(=S(2) Then Wh=(T'gross-S(1))*(S(3)*0.01)
220   If T'gross)S(2) And T'gross(=S(4) Then  Do
225     Wh=S(5)+((T'gross-S(2))*(S(6)*0.01)) : Enddo
230   If T'gross)S(4) And T'gross(=S(7) Then  Do
235     Wh=S(8)+((T'gross-S(4))*(S(9)*0.01)) : Enddo
240   If T'gross)S(7) And T'gross(=S(10) Then  Do
245     Wh=S(11)+((T'gross-S(7))*(S(12)*0.01)) : Enddo
250   If T'gross)S(10) And T'gross(=S(13) Then  Do
255     Wh=S(14)+((T'gross-S(10))*(S(15)*0.01)) : Enddo
260   If T'gross)S(13) And T'gross(=S(16) Then  Do
265     Wh=S(17)+((T'gross-S(13))*(S(18)*0.01)) : Enddo
270   If T'gross)S(16) Then Wh=S(19)+((T'gross-S(16))*(S(20)*0.01))
280   Return
```

Please note that this program is considerably shorter than the first. In Lines 220 through 265, I have used "If-Then Do-Enddo" to shorten the length of the lines for publication. You can actually use just "If-Then" as in Lines 270 and 210. This would shorten the program by 5 lines.

Even though both programs accomplish the same task, it is obvious that the latter shows better programming technique.

As always, if you have any comments, questions, or suggestions, please let me hear from you.

We're Interested!

We're Really Interested...in what you have to say. Especially about how you use your system...the problems encountered and the solutions effected...unusual uses or environments...and any practical applications you would be willing to share with fellow members. These can be short notes for departments like 'bits & bytes...' and 'Tec Tips,' or full feature articles.

Contact Bill Jaenicke at I/O News for editorial guidelines or assistance. We're interested in unleashing your literary talents.

THE HACKER'S HOME

THE HACKER'S HOME explores techniques in 'C' programming for Cromix and UNIX. Users are encouraged to submit utility programs of their own. It is edited by Rick Dhaenens, Senior Manager for the Cromemco, Inc. Atlanta Regional Office, 5901-C Peachtree Dunwoody Rd, Suite 375, Atlanta, GA 30328. (404) 391-9433.

Utilities for Profit

There are several utilities available under both UNIX and Cromix that can make life a lot easier for those of us who need them. These utilities are usually written by people to solve a particular problem or even just for fun.

One of the most popular computers of this day and age is the IBM PC. The PC has overrun the market not through it's technical abilities but through the availability of hundreds of application software packages. There are also portable PC clones that are useful for the traveler. The IBM PC disk format has become the *defacto* standard and is used for many non IBM machines. This presents a problem for those of us who would like to get data from or send data to a PC user.

MSCOPY, a utility written by Steve Gross of Cromemco, is the best solution I have seen to this problem. MSCOPY is available for both the UNIX and Cromix + operating systems. MSCOPY is like the Cromix **cdoscopy** utility and will allow reading, writing and initialization of 8 or 9 sector PC-DOS or MS-DOS diskettes. MSCOPY has an extensive set of options that provides a complete solution to moving data between a Cromemco and a PC. PC disks can be initialized either single or double sided and with eight or nine sectors. The boot block can be read from one floppy and written to another. Files can be accessed by the full pathname with automatic conversions between the MS-DOS backslash and the UNIX-Cromix slash conventions. File attributes can be modified on the PC disk files. There is even a rename option for files on the PC disk. All of this is done with very good error checking and a multitude of error messages. This is one of the most complete implementations of a disk format conversion utility I have seen. MSCOPY is available for $95.00 for either Cromix + or UNIX. Dealer discounts are available for quantity orders. Steve may be contacted by 300 or 1200 baud modem at (415) 857-9208. Login as MSCOPY to request information or order the MSCOPY utility. The mailing address is:

Cipher Systems
P.O. Box 6105
Stanford, CA 94305

Once you can move information back and forth between your PC and Cromemco machine you may want to run some of the software on both machines. Charlie Dobson of Systems Atlanta has just the thing, Structured Basic for the PC. This is a full implementation of Cromemco's 32k Structured basic set up specifically for the IBM PC. There are several enhancements to Sbasic including a built in sort, formatted input, and the ability to call DOS programs from Sbasic. This would allow you to run the same Sbasic programs on your PC or portable that you currently run under Cromix. For more information contact Charlie Dobson at (404) 928-0240 or by mail at:

Systems Atlanta
P.O Box 99
Lebanon, GA 30146

My particular area of interest has always been computer graphics. There is an excellent paint utility available for users of Cromemco's S-series graphics boards. This is the **Cgraf** utility written by Keith Musson of Digital Analysis Corporation. Cgraf uses a bitpad as the user interface to the graphics boards and is similar in function to the Cromemco Slidemaster package that was available for the older SDI graphics boards. This is a basic set of utilities that allows the user to load, save and manipulate graphic images on the S-series hardware. This package is compatible with the Baseline graphics package and supports all Baseline fonts and has cut and paste capabilities as well as image rotations. Support for the hardware zoom and scroll features of the S-series boards is also included. Cgraf is completely menu driven and is very easy for even the novice user. Cgraf is currently available for the Cromix operating system and will soon be available for UNIX as well. For more information contact Randy Webster at (703) 476-5900 or by mail at:

Digital Analysis Corp.
1889 Preston White Dr.
Reston, VA 22091

I have tried all of these utilities and have been impressed by the quality of the software and the documentation that comes with them. I will keep my eyes and ears open for other useful programs and utilities and will share my comments and observations in future articles.

CGRAF

Interactive Graphics System

AN EASY-TO-USE SOFTWARE PACKAGE FOR CREATING DETAILED TECHNICAL DRAWINGS AND COLORFUL ARTISTIC IMAGES

Features:

- Zoom - temporarily enlarges the image size on the screen
- Pan - moves the displayed view port around the entire graphics universe.
- Color Change and Mix -- changes the color used in displaying and storing any text, shapes or new display segments.
- Icon - creates, saves and recalls icons.
- Load Font - stores up to four fonts in memory from an unlimited number of text fonts.
- Hardcopy - transfers foreground/background or monochrome images to a printer.
- Shapes - draw circles, rectangles and lines.
- External - executes external command files.
- Reflections - reflects an image segment around its center in the x or y axis.
- Rotation - performs a 90, 180 or 270 degree rotation on an image segment.

Contact: Randy Webster
Digital Analysis Corporation
1889 Preston White Drive
Reston, VA 22091

Phone: (703) 476-5900

CGRAF

Creates, modifies, stores, and recalls high resolution images in either NTSC (National Television Systems Committee) or PAL (Phase Alteration Line) format on Cromemco S Series Graphics Cards, with SummaGraphics or GTCO digitizers.

Creates and displays 256 colors simultaneously from a total palette of over 250,000 colors, hues and shades.

Prices:

NTSC - $1195
PAL - $1495

Dealer and distributor discounts available.

D A C

Digital Analysis Corporation

BITS & BYTES

BITS & BYTES is the place to look for the odd bit of information, opinion, programs, profiles and rumors that circulate through The IACU. Our ears are always attuned to any interesting miscellany — if you have something to contribute send it along to I/O NEWS • P.O. Box 17658 • Irvine, CA 92713 • (714) 661-9764

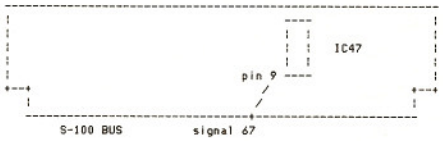
MACROTECH MEMORY BOARDS

In the last *BITS & BYTES* column we ran a query submitted by Mr. Vlaun regarding the use of MacroTech memory boards in Cromemco systems. IACU member Robert Turner submitted the following reply, which has been forwarded to Mr. Vlaun:

I have been using MacroTech International's MAX-M (one megabyte) memory board, rev. 1.0 with 68000 Cromix in a Cromemco System Three since May 1983 — I have one of the first production memory boards. I am using the board in the 24-bit address mode, and do not have the 16-bit M3 memory mapped addressing option.

There are two hardware modifications required to properly implement this board in a Cromemco system:

1. 16FDC Board — install a jumper on the circuit side of the board (not the component side) from pin 9 shown below on IC47 to signal 67 on the S-100 bus.



2. DPU Board — mask out signal 67 on the S-100 bus. This can be accomplished by using a piece of masking tape.

That's it. The operating system should boot and recognize all the memory available.

Robert W. Turner, P.E.
P.O. Box 761
Wethersfield, CT 06109
(203) 665-0211

ANNOUNCEMENT

Advanced Computer Software, Inc., (ACS) has been contracted to modify and upgrade the SASI bookkeeping system. We would like your comments to help us select the most appropriate changes. We would like to know, from both individuals and dealers, if you have an interest in this product, problems you have encountered with other products, and target systems (68000, Z80, MS-DOS). specifically for hard disk CROMIX systems, using Cromemco's Structured Basic. The design and direction was under the control of a CPA with many years in the accounting field, as well as a long stint with the IRS. The initial installation was at an accounting firm with over 100 clients, with several operators processing simultaneously. The first SASI ads appeared in I/O NEWS in 1980, so the product has been around for some time. The major efforts in the past three years have been aimed at reducing processing time to a minimum, allowing a bookkeeping operation to substantially reduce overhead.

The system was designed to be the backbone of the bookkeeping/accounting function, It does not provide complete accounts payable/receivable or payroll functions, which are available separately. It does, however, meet the needs of most businesses, particularly cash basis businesses and individuals. Departmental accounting is supported. SASI maintains detail for an entire year. This allows reports to be run at any time. Reports include general ledger, balance sheet, income statements, schedules, funds statements, payroll information, and vendor information. The optional Trial Balance Review (TBR) module allows accountants and internal auditors to review data and process tax return information while normal daily processing continues. The name SASI is derived from Satellite Accounting Systems, which is based on a main office with remote "satellite" operating systems. This method of distributing the processing further reduces the accountant's overhead. Through use of this system, Bob Strathearn, CPA, Irvine, California, has reduced his overhead by nearly $100,000 per year, with approximately the same number of clients.

Planned changes include further streamlining of menus and screens, porting to 68000 Structured Basic, and porting to MS-DOS.

For further information:

Pat McGuire
Advanced Computer Software, Inc.
700 S. Tustin Avenue, Suite B
Orange, CA 92666
(714) 771-0852

UUCP DOCUMENTATION

Cromemco recently released a 27 page Technical Bulletin supplementing the existing UNIX System V documentation of the communication programs CU and UUCP. CU (call another UNIX system) can be used to call or connect to another UNIX computer system. Once connected, CU can execute commands on that system as if your local terminal were directly connected to that computer. UUCP (UNIX to UNIX copy) is a program that copies files from one UNIX computer to another. Once properly set up, it will automatically copy files or send mail from one computer to another. Files can be queued for automatic transfer at specific times by telephone.

The technical bulletin details use of CU and UUCP between two computers using a direct connection as well as using a telephone connection to set-up automatic calling from a local computer to a remote computer.

It is too long to reproduce here in its entirety. However, copies are available for the asking. Contact *I/O NEWS* if you are interested.

68000 WriteMaster IN BETA TESTING

The project undertaken by QUINN TEAM to rewrite WriteMaster for the 68000 is well underway. A preliminary version has been completed and is under review. Beta testing is anticipated to begin in mid-May; a release version can be expected some time in June.

The announcement of this project met with a very favorable response. Among those that responded was Jeff Johnson, the designer and author of WriteMaster. He contributed the letter which follows:

Dear Mr. Quinn:

I noticed in *I/O NEWS* that you are considering rewriting WriteMaster for the 68000 family of Cromemco machines. As the designer and author of WriteMaster, I am flattered that someone thinks enough of it to want to reincarnate it on a more up-to-date system. I thought that I would share with you some of my thoughts about what I would do differently now that I know better.

Since I don't know whether you plan to look at the Z-80 code and program documentation when writing the new version (presumably in C) or just work from what the program does, some of my comments about WriteMaster internals may be superfluous. Also, my comments assume that your are somewhat flexible when you say "rewrite WriteMaster," in that some of my suggestions involve major departures from the way the program now works from the users' point of view. Finally, I am assuming that what you will be working from is WriteMaster 2.02, which was released two years ago (after leaving Cromemco, I fixed some bugs and sped up alignment, producing 2.03, but I don't think Cromemco has gotten around to releasing it yet).

1. Line breaking, alignment, and justification should be a display and printing function, rather than an operation that changes the text buffer. Alignment now works by adding and deleting soft Spaces and NewLines, which is very costly in time. The internalized text should be a sequence of characters separated by NewParagraph characters. No NewLines; that should be done at display and print time.

2. Text property toggles (e.g., boldface, underline) surrounding the affected characters in the file are bogus. Get rid of them. They complicate the buffer movement and display software tremendously. Instead, include data at the beginning of each paragraph regarding where character properties are turned on and off in that paragraph, and let the display and printing software worry about getting it right.

3. Have paragraphs remember their alignment properties. Current WriteMaster is inconsistent: character properties stick to characters and document properties stick to documents, but paragraph properties are merely a function of how the system is set when the Alignment command is given.

4. Get rid of the Load and Dump commands. They exist because, due to the limited memory of Z-80 systems for which WriteMaster was designed, the MOVE and COPY keys can't handle large blocks of text. In a 68000, you don't have such a limitation. Load/Dump is similar in function to Write/Read anyway.

5. Combine the LeftMargin setting with the Print Displacement setting, and don't display the left margin of the screen: it's a waste of valuable screen space and, given how the alignment software works, it requires the insertion of lots of useless spaces into the file and thus slows alignment down. After I added the printing displacement setting, I found that I used it as the (main) left margin and set the real LeftMargin setting to zero so I'd have more screen space and so things would align faster. WordStar works this way and it's better in this respect.

6. Improve selection. Either leave it basically as it is, but improve the feedback by having the video inversion extend as the cursor is moved after the first press of SELECT, or change it so that, e.g., a single press of SELECT selects a character, a (quick) double press in the same place selects a word, a triple click a sentence, four clicks a paragraph, then back to a character. If you do the latter, you can get rid of all the Delete Mumble commands and just use a single Delete that deletes the selection (since going to work at Xerox, I've become a big fan of keeping the number of commands down, making selection easy, and having users indicate the object of the command via selection). You might also consider having separate SELECT and EXTEND SELECTION keys or the latter could be invoked via SHIFT-SELECT.

7. When rewriting Merge, totally ignore what's there. It's terrible code that we never had time to rewrite.

8. Forget about Scan. I don't think anyone knew it was there.

9. Either figure out how to make the Find command special function keys (e.g., for specifying sets, ellipsis, etc.) more accessible/salient to users or get rid of them. Few people ever use them. If you keep them, add the capability to refer to characters and substrings matched by set-specifiers and ellipsis-specifiers in the replacement string (all the fancy pattern-matching stuff isn't very useful without this).

10. Read the message file in at the beginning and access it in memory instead of from disk. The message file idea is sound, because it vastly simplifies translation to other languages, but having the messages accessed from disk at runtime is painfully slow, especially from floppy.

11. Handle insertions and deletions via a linked list method (like Cromemco's new CE editor) instead of by moving the entire buffer apart and together constantly. Reorder the text when it gets too screwed up or at the end of the session.

12. Try to reduce the number of modes. When I became aware of the problem that modes cause for users (What mode am I in? Why did the mumble key do something different now than it did a minute ago? Why can't I do mumble in this state?), I began eradicating them from WriteMaster, but never finished the job. The LeftMargin key used to put WriteMaster into a mode (pre 2.00). Now it's modeless: its effect is based upon what the user does before pressing it, instead of after. The Print command used to put users through a series of modes/questions: users could either go through the sequence in order or ESCAPE out totally. Now it's less modal: it presents a form that allows the user to change any setting by moving the cursor. One example of a mode that remains is Overstrike vs. Insert. These days, I'd make Insert the default and simply not have Overstrike (users can delete and insert). Overstrike was made the default because I wanted WriteMaster to be similar to a typewriter for naive users, but I realize now that that was naive: WriteMaster ain't nothin' like a typewriter to nobody, not mention ill-advised: what's so great about typewriters?

That's all I can think of at the moment. If you actually go through with this and have further questions, feel free to contact me.

I should say that while I am flattered by this, I am not sure it's worth it. There are so many good third-generation word-processors out there these days, I'm not sure that keeping a second-generation word-processor (first generation was Electric Pencil, Screen [8] Formatter, etc.; second is WriteMaster, WordStar, VolksWriter, etc.; third is MacWrite, Word, etc.; fourth is what I'm working on now) alive will be worth it. Cromemco should buy one of the third-generation ones to sell with their systems.

Good luck!
Sincerely,
Jeff Johnson
Xerox Information Systems Division
2100 Geng Road
Palo Alto, CA 94303

dBASE II UPDATE

Most Cromemco users that use Ashton Tate's dBASE II are probably using version 2.3B, which was the first version that would run under CDOS and Cromix. There is a more recent version that also works: version 2.43*. It is available to registered dBASE II users from Ashton-Tate for a nominal upgrade fee. There is a hitch: it is only available on 8" standard format (CP/M) SS/SD diskettes.

According to Norman Vadnais, of Computer Specialists & Associates, many of the troublesome bugs present in 2.3B have been eliminated. He also cautions that the newer version is not as forgiving: in some instances 2.3B would accept syntactically incorrect statements. Therefore, existing command files may report errors when run under 2.43*.

Vadnais also stated that his company is working with Ashton-Tate to port dBASE III to Cromix-Plus. *Continued*

NEW PRODUCTS...

NEW PRODUCTS is a regularly appearing column devoted to announcing and following hardware and software products of interest to Cromemco users. Most information is derived from press releases submitted by vendors. As a result, I/O NEWS cannot be responsible for errors of ommission or any other inaccuracies.

New APPGEN Release

Software Express has released a new version of APPGEN applications that is faster than similar applications written in COBOL. This new APPGEN release, Version 1.6, is the first 4th generation language in the UNIX marketplace to produce applications surpassing the performance level of similar COBOL-based applications.

Using benchmark studies against typical COBOL applications, APPGEN applications ran an average of 20% faster. This speed improvement applies to any application created using the APPGEN Development System. This includes nine APPGEN-based accounting and business applications and 75 vertical market applications from APPGEN software houses.

In addition to a dramatic performance improvement, the new version of APPGEN features an easy-to-use installation program. By selecting standard default installation definitions, the APPGEN application is automatically installed. This provides the VAR with a simple, straightforward way to ensure accurate and fast software installation. At the same time, the flexibility of the installation procedures gives the VAR the option to customize the installation if necessary.

The new version of APPGEN also features a set of newly released Training Tutorials. Working with the Tutorials, end users can be trained on the new system in days instead of weeks. Thus, users are trained quickly yet at their own pace, and the VAR's time is devoted to new system sales instead of training.

The new version of APPGEN Development and Applications is compatible with all previous versions. Current APPGEN users can upgrade to the new version at no charge under a yearly software support agreement.

APPGEN Development sells for $6,000 while most APPGEN applications are $600 for a wide variety of mid-range machines. The Tutorials are retail priced at $125. The entire APPGEN product line, Version 1.6, is immediately available for over 50 UNIX and XENIX machines. For more information contact:

Linda L. Duttenhaver
The Software Express
2925 Briarpark Drive, 7th Floor
Houston, Texas 77042
(800) 231-0062
(713) 974-2298

Arstar II and M.CAD

Cromemco, Inc. has recently entered into two joint marketing efforts combining their supermicrocomputers with specialized graphics software: **Artstar II** and **M.CAD**.

Artstar II, developed by ColorGraphics Systems, Inc. of Madison, Wisconsin, is a highly sophisticated computer paint system that provides an exceptionally high-resolution medium for creating commercials, illustrations and other production visuals. It offers full compatibility with NTSC and PAL video standards and the ability to display 16 million colors. Additionally, it provides five different types of animation, generates anti-aliased type and produces such nuances as chrome effects, glows and gleams. It performs full-color frame grabs, generates backgrounds and offers fully variable on-screen air-brush capability. At the heart of ArtStar II is Cromemco's **System 400** equipped with a **Maximizer**. Cromemco dealers specializing in sales and support of the ArtStar II system include Circuit Studios, Inc. Betheseda, MD; Interactive Picture Systems Inc., New York City; Odyssey Systems Inc., Cleveland; MSC Electronics Ltd., Richmond Hill, Ont., Canada; and Micro Systems A/S, Oslo, Norway.

M.CAD is an advanced 3D mechanical Computer Aided Design and Computer Aided Manufacturing (CAD/CAM) system designed by Microcad, of West Hartford, CT. It is a highly sophisticated computer workstation which provides a full featured, integrated CAD/CAM system ideal for the parts production industry, small and medium tool and die shops, mechanical design, and Architecture, Engineering and Construction (AEC) markets. Utilizing Cromemco's **System 100**, M.CAD offers such benefits as:

- CAD/CAM on the same data base,
- Surface modeling with sculptured surfaces and blends,
- Catalogued subassemblies with associativity,
- An Informix data base for technical documentation and report generation,
- Advanced numerical control (NC) generation with easy editing,
- Comprehensive drafting package that supports both ANSI and ISO standards,
- General postprocessor for customization of NC.

Cromemco dealers specializing in sales and support of the M.CAD system include: Logic Corporation, Baton Rouge, LA;

Bits & Bytes

GOOD READING

Anyone who has spent time with a computer knows of the "interesting" things that can happen and of the frustrations which seem to lurk just around the corner. They may also know of the thrill which comes when you succeed in overcoming an obstacle. Ben Ross Schneider, Jr., an English Professor at Lawrence University in Wisconsin and a long-time IACU member, opened the door for all of this when, in 1980, he purchased a Cromemco Z-2D computer and set out to convert SITAR — a wordprocessing program developed on Lawrence University's PDP-11 computer — to run on the Cromemco. And therein is a story, one he reveals in his book, *My Personal Computer And Other Family Crisis*, (published by MacMillan).

The book is more akin to an adventure novel than anything else (although he does share a good deal of the technical know-how he gained). It's more philosophical than technical, and humorous to boot. As befits a Professor of English, the writing style is both fluent and precise, and shows a command of the language not usually found in books having to do with computers. It's entertaining and enlightening, but most of all it's just plain fun to read. I highly recommend it as an addition to your own computer library.

Ed.

ITI/X.25 USERS?

We've received a number of requests for help in setting up an ITI/X.25 network. If any of you have experience with this product, please contact I/O NEWS. You're help in this matter will be greatly appreciated.

NEW USER'S GROUP

I.A.C.U. member Mickey Stratton is interested in forming a local user's group in his area. If interested, he may be contacted at:

TAMKO Asphalt Products, Inc.
P.O. Box 2149
Tuscaloosa, AL 35403
(205) 752-3555 ext. 38

Computer Crossroads of America, Inc., Richardson, TX; Figgie Systems Management Group, Inc., Willoughby, OH; Accountability Systems, Orange, CA; Command GmbH, Ettingen, West Germany; Rocomp BV, Eindhoven, Holland; Micro Systems, Oslo, Norway; Consorzio Nazionale per L'Informatica, Bologna, Italy; Nescad Systems AG, Baden-Daettwil, Switzerland; Algorithm, Ltd., Athens, Greece; and N.C. Computer System Pty, Ltd., West Heidelberg, Australia.

CGRAF Interactive Graphics System

CGRAF is a hierarchical, interactive, fully menu driven graphics generating package. CGRAF is specifically developed for use on Cromemco 68000-based systems running Cromix D or Cromix-Plus. This software package takes full advantage of the graphic capabilities supported by the Cromemco S-Series graphics boards.

CGRAF is a user friendly software package. Detailed technical drawings as well as colorful artistic images can be created with minimum training. Previously digitized video images can be displayed and modified.

CGRAF creates, modifies, stores and recalls high resolution images in either NTSC (National Television Systems Committee) or PAL (Phase Alternation Line) format. The images created are multicolored full-screen images or image segments (ICONS). Displayed images have a resolution of 756 x 484 pixels on the NTSC hardware out of a total image size of 1024 x 512 pixels. On the PAL hardware the resolution is 756 x 577 pixels.

CGRAF can create and display 256 colors simultaneously from a total palette of over 250,000 colors, hues and shades. All 256 colors, including background and foreground, can be modified by the user.

Features:

- Zoom — temporarily enlarges the image size on the screen.
- Pan — moves the displayed view port around the entire graphics universe.
- Color change and mix — changes the color used in displaying and storing any text, shapes or new display segments.
- Icon — creates, saves and recalls icons.
- Load Font — stores up to four fonts in memory from an unlimited number of text fonts.
- Hardcopy — transfers foreground/background or monochrome images to a printer.
- Shapes - draws circles, rectangles and lines.
- External - executes external command files.
- Reflections — reflects an image segment around its center in the x or y axis.
- Rotation — performs a 90, 180 or 270 degree rotation on an image segment.

System Requirements:

1. Cromemco computer system with a minimum of 512K bytes of RAM.
2. Cromemco Tuart, Quadart or Octart board.
3. Cromemco S-Series graphics boards:
   Color Video Generator Board, SVID
   Video Memory Controller, SDMA
   2 Color Memory Boards, 256 KTP
   Baseline Graphics Development Software
   Color Modulator Board, SDCM (optional).
4. GTCO Digitizer (or other summagraphics compatible digitizer tablet).
5. Color or gray scale video monitor (RGB inputs are required unless the user has an optional Color Modulator Board in which case a good quality NTSC composite or PAL video monitor will do).

Prices:

NTSC —	$1195
PAL —	$1495

Dealer and Distributor discounts available.

Contact:

Randy Webster
Digital Analysis Corporation
1889 Preston White Drive
Reston, VA 22091
(703) 476-5900

Computer/Terminal Covers

If you've been looking for some nice covers for your Cromemco equipment, then here's some good news. **CompuCover** offers the following line of appropriate covers:

STOCK -	ITEM	PRICE
08500	Cromemco System Z2 System III	19.95
06600	Cromemco 3703, 3704	17.95
08700	Cromemco 3100 Terminal	19.95
08800	Cromemco 3102 Terminal	18.95
06800	Cromemco 3779 Printer	16.95
06900	Cromemco 3779 with Rack	16.95
06910	Cromemco C-5/C-10 CRT & keyboard	16.95
06920	Cromemco C-10 Disk stacked	8.95
06921	Cromemco CS-300	15.95
06922	Cromemco CTD Tape Drive	14.95

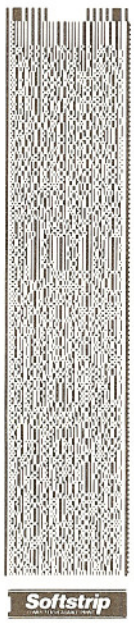
and others ...

For more information or to place orders contact:

CompuCover
P.O. Box 310
Mary Esther, FL 32569
(904) 243-5793
Telex 469783

Cauzin SoftStrip

Every now and then I hear about something really exciting. Although it is not yet available for Cromemco Systems, the Cauzin SoftStrip qualifies. The inset at the right is an example: encoded on the paper strip are three programs that appeared in Nibble Magazine demonstrating text, sound, and hires graphics. All that is needed to read the strip is the Cauzin Softstrip reader (which sells for $199) and the appropriate driver software (not yet developed for Cromemco systems).



The SoftStrip allows any type of digitally encoded information, be it text or data or program, to be represented in a readable format on a strip of paper. The ramifications of this technology are significant. Imagine that all of you had such a strip reader. Then, in addition to providing the printed listing of a given program in I/O NEWS, the SoftStrip encoded version could appear along side of it. Instead of having to key in the program you could read it directly into your computer using the Cauzin Strip Reader. No more modem transfers or mailing floppy disks. It would be great.

I've spoken with the folks at Cauzin, and they say that developing the software shouldn't be too big of a deal. They also informed me that they were working on software that would allow users to generate their own strips using a dot-matrix printer. There would be different versions tuned to reproduction by photocopying or by camera.

I think that this technology definitely holds great promise. It's one of those situations where if enough Cromemco users let it be known that they are interested, the system might become available (and not be limited to PC and Mac users). So let's all get on the ball and let them know that we are interested (and send a copy of your letter to I/O NEWS so that we can compile a list of interested Cromemco users).

Contact:

Cauzin Systems, Inc.
835 South Main Street
Waterbury, CT 06706
(800) 533-7323

LOCAL USER GROUPS

Assisting in the formation of Local Cromemco User Groups is one of the services performed by The IACU. We will be happy to provide you with a list of our members in your area, and recommend other contacts to help you organize and maintain a Cromemco computer users group. Just call or write I/O NEWS.

Arizona Association of Cromemco Users
Contact: Jo Ann Drake, President
2207 West Eugie Avenue
Phoenix, AZ 85029
(602) 993-9589

Australia User's Group*
Contact: Minicomp
Minicomp Building
104 Mount Street
North Sydney, NSW 2060
Australia
(02) 957-6800
Meets Monthly
*Publishes "Minicomp/Cromemco" a monthly newsletter

Bay Area Cromemco Users & Programmers (BACUP)
Contact: Raymond Barglow or Alan Walworth
United Word & Data Processing
2345 Fulton Street
Berkeley, CA 94704
(415) 841-0708 or (415) 548-2692

Cromemcohorts
Contact: Dr. Brent Lowensohn
4747 Sunset Blvd.
Los Angeles, CA 90027
(213) 667-8972

Cromemco Users' Group of Australia*
Contact: Tony Stringer
52 Beechwood Avenue
Greystanes, 2145
*Publishes a magazine "CROME-SOMA"

Cromemco Users' Group Holland (CUGH)
Contact: Joop Kohler, Secretary
P.O. Box 120
2910 AC Kieuverkerk a/d IJssel
The Netherlands 01803-13300

Cromemco Users' Group*
Contact: Peter Norman
The University of Newcastle Upon Tyne
Department of Chemical Engineering
Merz Court, Claremont Road
Newcastle Upon Tyne NE1 7RU
England
Newcastle 28511, Ext. 3278
*Publishes Cromemco Users' Newsletter (CUG)

Cromemco Users' Group Ontario, Canada
Contact: Lloyd Parker
Hiram Walker Resources, Ltd.
Suite 600
1 First Cnandian Place
Toronto, Ontario
Canada M5X 1A9
(416) 864-3349

Cromemco Users of Orange County, California
Contact: Michael Peterson
Accountability Systems
700 South Tustin Avenue
Suite B
Orange, CA 92667
(714) 639-4570
Meets third Tuesday monthly

Insystems Pty. Ltd.*
Contact: Norman Rosenbaum
337 Moray Street
South Melbourne, Victoria
3205 Australia
(03) 690-2899, Telex: AA30458
*Publishes "Cromemco UPDATE" a bi-monthly newsletter

Illinois Users' Group
Contact: Jim Knowles
P.O. Box 631
Elgin, IL 60120
(312) 695-7775

Indonesian Cromemco Users' Group (ICUG)*
Contact: Zafir M.A. Pontoh
Computation Lab
Department of Regional & City Planning
Bandung Institute of Technology
10 Ganesha
Bandung, Indonesia
(022) 82051 ext. 360
*Publishes "BERKALA ICUG" a monthly newsletter

Microcomputer Users' Group
Contact: Noble Bright
P.O. Box 1
Cape May, NJ 08204
(609) 884-2222
(609) 429-3838
Meets fourth Wednesday monthly

Northwest Association of Cromemco Users (NWACU)
Contact: Jim Illman
403 S. Brandon
Seattle, WA 98108
(206) 763-2099

North San Diego County Users' Group
Contact: Charles Mackey
P.O. Box 397
Fallbrook, CA 92028
(619) 728-6116
Located 30 mi. east of Oceanside

North Texas Cromemco Commercial Users' Group
Contact: Jerrell Johnson
1131 Winterwood
Lewisville, TX 75067
(214) 221-1437
Or call Rocky Hall
at (214) 398-1595
Meets first Wednesday bi-monthly

NY, NY Users' Group
Contact: Charles Perrella
45F Route 303
Valley Cottage, NY 10989
(914) 268-5137

SaCromemco Users
Contact: Alan Whitman
Box 244
Rancho Cordova, CA 95670
(916) 635-6070

Silicon Valley Cromemco Users
Contact: Allan O'Neill
(415) 969-3854 or Emily Ott (415) 854-5818
meeting place provided by:
MCM Enterprises
215 Hamilton Avenue
Palo Alto, CA 94301
Meets fourth Tuesday monthly

W.A. Cromemco Users' Group
Contact: Rae Canning
c/o The W.A. School of Computing
2/294, Rokeby Road
Subiaco, Western Australia 6008

West Germany Users' Group
Contact: Glynnis Long
Tesco GmbH
P.O. Box 10
8714 Weisentheid
West Germany
09383-1237
Total fluency in English & German

Wisconsin Cromemco Users' Group
Contact: Bob Ungemach
6249 West Browndeer Road
Browndeer, WI 53223
(414) 355-1451

The International Association of Cromemco Users

Software Resource Survey

— PURPOSE —

The purpose of this survey is to further expand the list of known software applications running on Cromemco systems. By taking the time to complete and return the questionaire you will be doing yourself and Cromemco users worldwide a tremendous favor. Everyone needs software, but finding the package you need can be terribly frustrating. This survey could eliminate that frustration. So please, help us help you by returning the completed questionaire at your earliest convenience.

Thank you!

— INSTRUCTIONS —

There are three (3) sections to this survey. Put your name and address information in the USER section. Use the SYSTEM section to describe the Cromemco system on which the software is run. If you have different systems running different operating systems and software, photocopy the survey sheet and submit one for each system.

The survey sheet has space for four (4) software package descriptions. If you have more than that, make some additional photocopies. For each package, provide the following information: the **product name**, a brief **description** of the software (spreadsheet, wordprocessor, DBMS, etc.); if you had **problems** in getting the package up and running, so indicate with comment; indicate whether or not you are satisfied with the package. Also, please provide the address and telephone information about the software **vendor** or manufacturer so that we can submit more detailed Product Fact Sheets to them. If the application was designed in-house, please make note of it and indicate if you're interested in having the application included in the **IACU Software Resource Guide**. If so, we'll send you some Product Fact Sheets.

When completed, please mail the survey sheets to:

The I.A.C.U.
P.O. Box 17658
Irvine, CA 92713

USER

Name _____ Member No._____

Address _____

_____

City _____ State _____ Zip _____ Phone _____

Country _____

SYSTEM

Model: _____

Memory: _____

Disks: _____

Operating System: ☐ CP/M ☐ CDOS ☐ Cromix

☐ Cromix-D ☐ Cromix-Plus ☐ UNIX ☐ Other_____

SOFTWARE

Product Name: _____

Description: _____

Problem? ☐ Yes ☐ No _____

Like it? ☐ Yes ☐ No _____

Product Name: _____

Description: _____

Problem? ☐ Yes ☐ No _____

Like it? ☐ Yes ☐ No _____

Product Name: _____

Description: _____

Problem? ☐ Yes ☐ No _____

Like it? ☐ Yes ☐ No _____

Product Name: _____

Description: _____

Problem? ☐ Yes ☐ No _____

Like it? ☐ Yes ☐ No _____

VENDOR INFO

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

BEST IN SALES AND SERVICE

We understand your desire to have a local Cromemco dealer in your area, and we believe we have a perfect solution for your problem at this time.

We are THE MOST QUALIFIED facility in the U.S. as we are the first recipient of Cromemco's new award "Number 1 U.S. Dealer". Again we have received awards for "Outstanding Service" as well as ranking in the "Top 25 Dealers". Based in the greater Dallas area, we sell and service Cromemco equipment in all regions of the U.S., and in several foreign countries.

Due to our central location and the convenient access to DFW Airport, equipment can be received from any location in the U.S. and returned to the customer within three working days (excluding transit time), if our expedited service is requested.

Service and support are the reasons that we are successful at what we do. Simply said, sales backed by support means success and that means **Computer Crossroads**.

SPECIALS OF THE MONTH

HARD DISK

MICROPOLIS 1304 50MB.....List $2100.00.....1595.00
SEAGATE ST 225 20 MB.....List $ 420.00..... 365.00

FLOPPY DISK

PANASONIC JU455-3AA 1/2 HGT 5 1/4" LIST $250..... 125.00
TANDON TM848-2E 1/2 HGT 8" LIST $695..... 450.00

BOARDS

512 MSU (WHILE SUPPLY LASTS) LIST $485.00..... 375.00
256KZ (WHILE SUPPLY LASTS) LIST $795.00..... 520.00

MISCELLANEOUS

WY-30 WYSE TERMINAL 14" FLAT SCREEN..... 320.00
DETACHABLE KEYBOARD/41 PROGRAMABLE KEYS
2400 - 2400 BAUD U.S. ROBOTICS MODEM..... 432.00
UPS 800 CAP UNINTERRUPTABLE POWER SUPPLY..... 799
POWER STRIP 6 OUTLETS/FUSE/SURGE..... 14.95

COMPUTER CROSSROADS of AMERICA, INC.

SOFTWARE
SERVICE
HARDWARE

Where It All Comes Together

We will meet or beat any bonafide sales quote on Cromemco equipment and will make available the best in support at no extra cost.

6 Terrace Shopping Center • Richardson, Texas 75081

FOR SUPER SERVICE CALL (214)231-6108 Telex: 4991118

SYSTEMS ATLANTA, INC.

CROMIX SLAVE PROCESSORS

8MHZ Z80H 256K 2 Serial Ports

Two processes per slave

FAST – FAST – FAST – FAST – FAST

Eliminate degradation of system performance when running CP/M programs like dBASE and WORDSTAR in multi- user environments

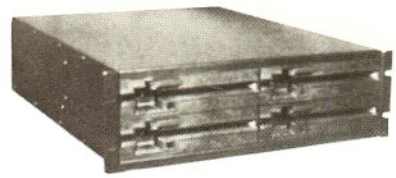
Runs under Cromix 11.27 and Cromix-Plus

$1095.00 including software license

Custom Disk Drive Cabinets and Power Supplies for Hard Disks, Cartridge Tapes, 5.25″ and 8″ floppies

Desk Top and Rack Mount

Call or write for information and prices



NEW 32K STRUCTURED BASIC

CROMIX (SBASIC.BIN)	$295
PC/MS DOS	$295
CP/M	$195
UPDATE CROMIX VERSION	$ 95 *
UPDATE CDOS VERSION	$ 75 *

* Available to registered Cromemco Licensees only.

FAST BACKUP for CROMIX Hard Disks $95
Backup to floppy disk at one megabyte per minute, 8″ or 5.25″

FAST diskette copy programs $95
Copy both 8″ and 5.25″ CDOS, CROMIX and UNIX formats

Z80 cartridge tape backup utility $95
(Included with purchase of CTD tape subsystem at $1295)

CALL OR WRITE FOR INFORMATION ON ALL PRODUCTS AND SERVICES.

SYSTEMS ATLANTA, INC., P.O. BOX 99, LEBANON, GA 30146 (404) 928-0240

CROMEMCO COMPUTERS: DESIGNED TO MAKE UNIX SYSTEM V EVEN BETTER...

UNIX System V, the new standard in multi-user microcomputer operating systems, gives you high performance features along with the portability and flexibility of a standard.

Cromemco computers can make UNIX System V even better. Because our systems are designed with UNIX in mind. First of all, we offer UNIX System V with Berkeley enhancements. Then, our hardware uses advanced features like 64K of on-board cache memory and our high speed STDC controller to speed up disk operations – very important with UNIX.

More capability and expandability

We have a high-speed, 68000-based CPU that runs at 10 MHz, coupled with a memory manager that uses demand-paging and scatter loading to work *with* UNIX, not for it.

We provide room for expanding RAM to 16 megabytes – with error detection and correction – for running even the most sophisticated and advanced microcomputer programs. And the power to accommodate up to 16 users – all with plenty of memory.

But we give you even more.

A complete solution

We give you a choice in systems: the System 100 series, expandable up to 4 megabytes of RAM, and the System 300 series, expandable to 16 megabytes. A high speed 50 megabyte hard disk drive is standard on the systems. And you can expand the hard disk capacity up to 1200 megabytes using standard SMD drives. You can add floating point processing. High resolution graphics. Video digitizing and imaging. Communications through standard protocols. Mainframe interface.

And software support is here to meet your needs. We offer major programming languages, database management systems, communications software, including SNA architecture, X.25 protocol, and Ethernet; even a program to interface to an IBM PC if you need to. And, of course, access to the broad range of standard UNIX applications programs that is growing dramatically every day.

Easy to use.

We also make our systems easier to use, because we install the operating system before we ship your computer. No complicated installation procedures. And the Berkeley enhancements give you the standard UNIX System V operating system, but with the added convenience of these widely acclaimed improvements.

Cromemco's System 100 and System 300 computers: designed to be the highest performance UNIX systems available anywhere.

Just call or visit one of our UNIX System V Official System Centers to see for yourself. They'll also give you a copy of our new publication, "What you should know before you buy a UNIX system." Or contact us directly.

We'll be glad to show you how to get a better UNIX system.

Corporate Headquarters: Cromemco, Inc., 280 Bernardo Avenue, P.O. Box 7400, Mountain View, CA 94039. (415) 969-4710. In Europe: Cromemco GmbH, 6236 Eschborn 1, Frankfurter Str. 33-35, P.O. 5267, Frankfurt Main, Germany.

UNIX is a trademark of Bell Laboratories.
IBM is a trademark of International Business Machines Corp.

Cromemco®